

Juha Hulkkonen

## **VIRTUALISOIDUN TESTAUSYMPÄRISTÖN RAKENTAMINEN**

Opinnäytetyö  
Kajaanin ammattikorkeakoulu  
Luonnontieteiden ala  
Tietojenkäsittely  
Kevät 2015

Koulutusala Luonnontieteiden ala	Koulutusohjelma Tietojenkäsittely
Tekijä(t) Hulkkonen, Juha	
Työn nimi Virtualisoidun testausympäristön rakentaminen	
Vaihtoehtoiset ammattiopinnot	Toimeksiantaja CSC – Tieteen tietotekniikan keskus Oy
Aika Kevät 2015	Sivumäärä ja liitteet 51
<p>Testaaminen on olennainen osa uusien tietojärjestelmien ja sovellusten käyttöönottoa. Yritysten tuotantoympäristöissä on yleensä useita kriittisiä sovelluksia ja palvelimia, joiden toimintaympäristö on pidettävä mahdollisimman vakaana. Uuden, ennalta tuntemattoman sovelluksen, palvelun tai verkkoratkaisun käyttöönotto tällaisessa kriittisessä ympäristössä voi aiheuttaa ongelmia niin omille kuin mahdollisille asiakkaiden palveluille.</p> <p>Uudet sovellukset ja palvelut tulee testata huolellisesti tähän tarkoitukseen varatussa testausympäristössä. Testausympäristö on erotettava yrityksen tuotantoympäristöistä. Tietoverkkoja käyttävät huonosti toimivat palvelut voisivat samassa ympäristössä toimiessaan hidastaa ja pahimmillaan estää tuotantoympäristön palveluiden toimimista.</p> <p>Tässä opinnäytetyössä määriteltiin, suunniteltiin ja toteutettiin asiakkaalle virtualisoitu testausympäristö. Työ toteutettiin toimeksiantajan aikaisempien virtualisointiympäristöjen tapaan VMwaren virtualisointituotteilla. Työn lopputulos on täysin käyttövalmis ja dokumentoitu järjestelmä sovellusten, palveluiden ja verkkojen testaamiseen. Tässä dokumentissa kuvataan työn taustoja, suunnittelutyötä ja käytännön toteutusta.</p> <p>Työn tilaaja on CSC – Tieteen tietotekniikan keskus Oy, joka tarjoaa Kajaanin Renforsin Rannassa sijaitsevasta konesalista suurteholaskenta-, tiedontallennus- ja pilvipalveluita Suomen yliopistoille, korkeakouluille ja tutkimuslaitoksille. CSC on valtion omistama, voittoa tavoittelematon osakeyhtiö, jonka päätoimipiste sijaitsee Espoossa.</p>	
Kieli	Suomi
Asiasanat	Virtualisointi, VMware, Testaus
Säilytyspaikka	<input checked="" type="checkbox"/> Verkkokirjasto Theseus <input type="checkbox"/> Kajaanin ammattikorkeakoulun kirjasto

School Business	Degree Programme Business Information Technology
Author(s) Hulkkonen, Juha	
Title Creating Virtualized Testing Environment	
Optional Professional Studies	Commissioned by CSC – IT Center for Science Ltd.
Date Spring 2015	Total Number of Pages and Appendices 51
<p>Testing is an essential part of developing new IT-systems and applications. There are many critical applications and servers in the corporate production environment. This kind of environment needs to be stable. When commissioning new unknown applications, services or network solutions into a production system there may be problems that affect internal or customer services.</p> <p>New applications and services have to be fully tested in the special system dedicated testing environment. It should be separated from the production systems. Badly working services could slow down or even deny production services from working if they would run in the same environment.</p> <p>This thesis deals with defining, designing and building a virtualized testing environment for the client. The system was built up with VMware virtualization products which are widely used by the client. The result of the work was a fully operational and documented system for the testing of applications, services and networks. This document describes the background of the work, designing process and how the system was built up in practice.</p> <p>The commissioner for this thesis is CSC – IT Center for Science Ltd. offering high performance computing, data storage and cloud services from the Kajaani Data Center which is located in the Renforsin Ranta business park. Services are offered for the Finnish universities, universities of applied sciences and research institutes. CSC is a state owned non-profit limited company which headquarters is located in Espoo.</p>	
Language of Thesis      Finnish	
Keywords	Virtualization, VMware, Testing
Deposited at	<input checked="" type="checkbox"/> Electronic library Theseus <input type="checkbox"/> Library of Kajaani University of Applied Sciences

## SISÄLLYS

1 JOHDANTO	1
2 CSC	5
2.1 CSC yrityksenä	5
2.2 Kajaani DataCenter	6
2.3 Testiympäristön tarve CSC:llä	7
3 TESTAAMINEN	8
3.1 Testaamisen historiaa	9
3.2 Testaamisen lähtökohdat	9
3.3 Järjestelmätestaus	10
4 MÄÄRITTELY JA SUUNNITTELU	12
4.1 Vesiputousmalli	12
4.2 Testausympäristön teoriavaiheet	15
4.2.1 Esitutkimusvaihe	15
4.2.2 Määrittely	16
4.2.3 Suunnittelu	18
5 KÄYTÄNNÖN TOTEUTUS	30
5.1 Asennus ja asetukset	30
5.2 Testaus	44
5.3 Käyttöönotto ja ylläpito	49
6 POHDINTA	50
LÄHTEET	52

## SYMBOLILUETTELO

Best practices	Tarkoittaa parhaita käytäntöjä. Tällä ilmauksella kuvataan laitteen tai ohjelmiston toimittajan kertomia parhaita käytäntöjä laitteen tai ohjelmiston käyttöönottoon ja asetusten määrittämiseen.
CSC	CSC – Tieteen tietotekniikan keskus Oy
ESXi	VMwaren virtualisointipalvelin. Asennetaan fyysiseen palvelimeen. Ei vaadi toimiakseen erillistä käyttöjärjestelmän asennusta.
Host	Isäntäkone, virtualisointipalvelin, jossa ajetaan VMwaren tapauksessa ESXi-palvelinohjelmistoa.
iSCSI	Internet Small Computer System Interface. Teknologia, jolla liitetään tietokoneita levyjärjestelmään lähiverkon yli käyttäen TCP/IP-verkko-protokollaa. Ei vaadi erillistä tallennusverkkoa.
Klusteri	Usean <i>hostin</i> muodostama, yhtenä kokonaisuutena toimiva järjestelmä.
OpenSource	Vapaan lähdekoodin ohjelmisto. Ohjelmisto on vapaasti ladattavissa Internetistä ja lisenssiehdot sallivat ohjelmiston muokkaamisen.
LUN	Logical unit number. Käytetään identifioimaan loogista levyaluetta jaetussa levyjärjestelmässä. LUN näkyy sitä käyttävälle tietokoneelle yhtenä kiintolevynä, vaikka todellisuudessa se voi koostua useista eri kiintolevyistä.
RAID 1	Kiintolevyjen peilaus. Käyttää kahta kiintolevyä. Toiselle levyille kirjoitettu tieto kopioidaan automaattisesti toiselle, jolloin toisen kiintolevyn rikkoutuessa tietoa ei menetetä.
RAID 50	Yhdistää RAID 0:n tuoman nopeuden jakamalla kirjoitettavan tiedon useammalle RAID 5 levyryhmälle. Tallennettava tieto jaetaan usealle levyille ja tiedosta lasketaan pariteetti, joka tallennetaan erilliselle levyille. RAID 50 on nopea ja kestää yhden levyn hajoamisen yhtä levyryhmää kohden ilman tiedon menetystä.
RAID 6	Vikasietoisuutta parantava määrittely. Useasta kiintolevystä muodostettu levyalue, joiden lisäksi on kaksi pariteettilevyä. Levyalueelle jaetuista tiedoista lasketaan kaksi pariteettiä, jotka tallennetaan pariteettilevyille. Kestää kahden levyn hajoamisen ilman tiedon menetystä.
SAN	Storage Area Network. Tarkoittaa tietojärjestelmän palvelinten ja tallennusjärjestelmän väliseen tiedonsiirtoon käytettävä teknologia. Vaatii oman, vain tätä tarkoitusta varten luodun tietoverkon.
UPS	Uninterruptible Power Supply. Laite joka takaa tasaisen virransyötön lyhyissä sähkökatkoksissa ja tasoittaa syöttöjännitteen epätasaisuuksia.

vCenter Server	VMwaren työkalu virtuaaliympäristön keskitettyyn hallintaan. Muodostaa yksittäisistä ESXi-palvelimista klusterin.
Virtual appliance	Virtuaalikone, joka on räätälöity virtuaalilaitteiston ja ohjelmiston osalta valmiiksi pakettiksi tietynlaista tarkoitusta varten. Asentaminen käy helposti lataamalla levykuva virtuaaliympäristöön.
VMware	VMware Inc, virtualisointituotteita tekevä yhdysvaltalainen yritys

## 1 JOHDANTO

Tämän opinnäytetyön tavoitteena on perehtyä tietokoneohjelmistojen ja käyttöjärjestelmien testaamiseen tarkoitettun virtualisoidun tietojärjestelmän määrittelyyn, suunnitteluun ja toteutukseen. Lisäksi tavoitteena on oppia, mitä testaamisella tarkoitetaan uusien tietojärjestelmien ja sovellusten käyttöönotossa. Työn tilaaja on CSC - Tieteen tietotekniikan keskus Oy, jäljempänä CSC. Opinnäytetyön käytännön lopputulos on Kajaanin konesaliin pystytetty täysin toimiva, dokumentoitu ja käyttövalmis virtualisoitu ympäristö tietoteknisten sovellusten, käyttöjärjestelmien ja verkkojen testaamiseen. Kohdejärjestelmän käyttäjiä tulevat olemaan pääasiallisesti CSC:n Kajaanin konesalin työntekijät.

Tämä kirjallinen työ kertoo perehtymisestä testaamiseen sekä kokonaisen virtualisoidun tietojärjestelmän rakentamiseen. Työn myötä tutustutaan erilaisiin laite- ja sovellusratkaisuihin sekä verkko- ja tallennusjärjestelmiin, joita kyseisenlainen testausympäristö tarvitsee. Testausympäristön tarkoituksena on toimia CSC:n Kajaanin henkilöstön omana ympäristönä erilaisten käyttöjärjestelmien, verkkoratkaisuiden ja ohjelmistojen testaamiseen. Lisäksi ympäristöllä voidaan testata Espoon ja Kajaanin konesalien välisiä tietoliikenneyhteyksiä ja näitä yhteyksiä käyttäviä sovelluksia. Työssä tehdään käyttötarkvekartoitus, jonka myötä selviää tarkemmin mitä kaikkea järjestelmällä on tarkoitus pystyä testaamaan. Tämän tarkvekartoituksen tulos vaikuttaa järjestelmän laite-, ohjelmisto- ja asetusvalintoihin.

Tämän työn aikana haetaan tietoa testaamisesta ja tietojärjestelmistä yleisesti Kajaanin Ammattikorkeakoulun kirjastosta sekä perinteisistä kirjoista että elektronisista aineistoista. Internet tulee olemaan suuressa roolissa tiedonhaussa. Kirjallisten lähteiden lisäksi työssä käytetään etenkin määrittelyvaiheessa hyväksi CSC:n työntekijöiden haastatteluja.

Tietotekniikan alalla toimivan asiantuntijaorganisaation, jollainen CSC on, käytössä on hyvin paljon erilaisia ohjelmistoja sekä ohjelmistojen ja fyysisten laitteiden yhdessä muodostamia tietojärjestelmiä. Näitä ohjelmistoja ja tietojärjestelmiä käytetään sekä yrityksen sisäisten palveluiden ja prosessien että asiakkaille tarjottavien palveluiden tuottamiseen. Useat näistä ohjelmista ovat toimintavarmuutensa suhteen hyvin kriittisiä. Näin ollen ennen ohjelman tai palvelun tuotantokäyttöönottoa tulee se testata hyvin. Testaamisessa pyritään tarkistamaan ohjelman tai palvelun toimivuus erilaisilla asetuksilla ja kuormituksilla sekä yhdessä muiden sovellusten kanssa. Erilaisten tietoverkkojen ja verkkoasetusten kanssa toimimisen testaaminen on myös hyvin tärkeää.

Testausympäristöt ovat yleisesti erittäin tärkeässä osassa uusien sovellusten, järjestelmien ja asetusten käyttöönottoprojekteissa. Uuden sovelluksen käyttöönotto tai vanhaan sovellukseen tehtävien laajempien muutosten tekeminen muualla kuin turvallisessa, tuotantoympäristöstä eristetyssä ympäristössä voi olla erittäin riskialtista koko tuotantoympäristön toiminnan kannalta. Testattaessa puolivalmiita tai muuten epävarmasti käyttäytyviä laitteita tai sovelluksia on olemassa riski aiheuttaa haittaa muille samassa ympäristössä oleville laitteistoille tai palveluille. Järjestelmän ollessa tuotantokäytössä järjestelmän vikaantuminen voi olla erittäin kiusallinen ja jopa taloudellisesti merkittävä haitta niin omalle yritykselle kuin asiakkaillekin. Tuotantojärjestelmien huono tai epävakaa toiminta voi haitata paljon niin yrityksen omaa kuin asiakkaiden työntekoa. Huonosti toimivat sovellukset aiheuttavat ongelmia myös sovellusta tarjoavan yrityksen imagolle ja maineelle.

Tämän työn aikana tulee varmasti vastaan ongelmia ja huomioon otettavia asioita, joita ei ole osannut ajatella etukäteen. Näitä asioita tulee todennäköisesti olemaan eri laitteiden välillä tehtävissä valinnoissa, käytettävän virtualisointiohjelmiston valinnassa käyttötarkoituksen ja yrityksen yleisten ohjeistusten mukaisesti sekä asetusten määrittämisessä. Tallennusverkon teknologiaa joudutaan pohtimaan parhaan mahdollisen toiminnallisuuden aikaansaamiseksi, huomioiden kuitenkin kustannusten pysyminen maltillisina. Tulevia ongelmia pyritään vähentämään ennakolta huomioimalla laite- ja ohjelmistovalinnoissa yrityksen aiemmat käytännöt ja kokemukset sekä ottamalla huomioon ohjelmistojen, verkkojen ja tallennusjärjestelmän asennuksessa ja määrittämisessä valmistajien “best practices” eli parhaat käytännöt.

Kohdeympäristön rakenne suunnitellaan nykyaikaisten tietojärjestelmäkäytäntöjen mukaisesti joustavaksi ja helposti laajennettavaksi. Fyysiset palvelimet tullaan yhdistämään yhtenäiseksi ympäristöksi käyttäen virtualisointitekniologiaa. Suorituskykyvaatimusten kasvaessa tulevaisuudessa voidaan järjestelmää kasvattaa helposti lisäämällä uusia fyysisiä palvelimia. Uudet palvelimet voidaan lisätä samaan ympäristöön, jolloin ympäristön suorituskyky kasvaa prosessoritehon ja muistin määrän osalta. Virtuaalisen ympäristön tiedon tallennus hoidetaan kaikille fyysisille palvelimille yhteisellä levyjärjestelmällä. Yhteisellä levyjärjestelmällä mahdollistetaan laitteiston vikasietoisuuteen, kuormantasaukseen ja virransäästöön käytettävien toiminnallisuuksien hyödyntäminen. Erilliseen levyjärjestelmään voidaan lisätä levytilaa levy kerrallaan tarpeen mukaan. Kerralla suurempaa laajennustilaa tarvittaessa voidaan järjestelmään lisätä myös kokonaan uusi levyjärjestelmä. Levyjärjestelmää suunniteltaessa pidetään mielessä myös mahdollisuus käyttää hyväksi jo olemassa olevia levyjärjestelmiä, joista voisi lohkaista oman osion testausympäristön käyttöön. Tämä säästäisi kustannuksia, mutta toisaalta levyjärjestelmä



ei olisi siinä tapauksessa täysin testausympäristön oma ja näin ollen sen hallinta ja mahdollisesti joskus tarvittavat muutokset eivät olisi niin yksinkertaisia toteuttaa.

Työn kohdejärjestelmä toteutetaan toimeksiantajan aikaisempien virtualisointiympäristöjen tapaan käyttäen VMwaren virtualisointituotteita. VMware on tunnetuimpia markkinoilla olevia virtualisointituotteiden valmistajia. Toimeksiantajan henkilökunnalla on paljon tietotaitoa valmistajan tuotteiden käytöstä. Virtualisointituotteiden lisenssien hinnat ovat niin korkeita, että järjestelmää määritellessä selvitetään myös, olisiko osa ympäristön ohjelmistoista saatavilla ilmaisina OpenSource-tuotteina.

Virtualisoidun ympäristön isäntäkoneet eli *hostit* yhdistetään klusteriksi yhteisen hallintakoneen alle. Tämän hallintakoneen osalta tulee projektin aikana selvittää, löytyisikö tähän tarpeeseen soveltuvaa OpenSource-tuotetta, joka tarjoaisi tarvittavat toiminnallisuudet. Vaihtoehtoisesti hallintakone voisi olla myös VMwaren erillinen *virtual appliance* eli virtuaalinen valmissovellus, joka suorittaa hallintakoneen tehtävää. Mikäli muuta ratkaisua ei löydy, voidaan järjestelmän yhdistäminen hoitaa CSC:n olemassa olevaa vCenter-sovellusta käyttäen. Tällöin klusterin hallintasovellus toimisi Espoon konesalissa, eikä järjestelmä olisi näin ollen täysin Kajaanin henkilöstön hallinnassa.

Testausympäristö on luonnollisesti kytkettävä tietoverkkoon. Tähän opinnäytetyöhön ei kuulu järjestelmän ulkoinen verkkosuunnittelu, sillä järjestelmä kytketään CSC:n olemassa olevaan verkkoon ja eristetään tuotantojärjestelmistä. Virtualisointiympäristön sisäinen fyysinen tietoverkko ja virtuaaliset verkkoratkaisut järjestelmän ylläpitoon, kuormantasaukseen ja käyttöön kuuluvat kuitenkin tähän opinnäytetyöhön. Alusta alkaen lähtökohtana on, että järjestelmässä tulee olemaan ainakin kaksi erillistä verkkoa. Toinen verkoista toimii normaalina käyttöyhteytenä palveluihin ja palvelimille pääsyyn. Toinen verkoista on tarkoitettu pelkästään järjestelmän sisäiseen kuormantasaukseen ja ylläpitoon. Työn edetessä opiskellaan, olisiko ylläpitoverkko syytä erottaa kahdeksi erilliseksi verkoksi, toisen hoitaessa palvelinten välistä kuormantasausta ja toisen ollessa varattu pelkästään ylläpitotoimiin. Verkkoratkaisuihin ja käytännön ympäristöön toteutettavaan ratkaisuun perehdytään työn kuluessa tarkemmin.

Työ etenee aluksi VMwaren ohjedokumentteihin tutustumisella, testauksen teorian selvittelyllä ja niistä kirjoittamalla. Käytännön osuus alkaa järjestelmän käyttötarpeiden määrittelyllä, jonka jälkeen aletaan suunnittelemaan käyttötarpeiden perusteella, millaisilla ominaisuuksilla kyseiset tarpeet saadaan täytettyä. Suunnittelutyön valmistuttua päästään asentamaan, määrittämään

asetuksia ja testaamaan järjestelmää. Lopuksi pohditaan, kuinka projekti onnistui ja mitä olisi voinut tehdä toisin.

## 2 CSC

Työn tilaajana on CSC - Tieteen tietotekniikan keskus Oy (jäljempänä CSC) ja kohdeympäristön loppukäyttäjänä Kajaanin toimipisteen henkilöstö. Opinnäytetyön tekijä on työskennellyt CSC:llä Kajaanin toimipisteessä kesäkuun 2014 alusta alkaen aluksi kesäharjoittelijana ja myöhemmin nuorempana käyttöinsinöörinä. Tämän opinnäytetyön käytännön osuuden toteuttaminen kuuluu opinnäytetyön tekijän työtehtäviin muiden CSC:llä hoidettavien velvollisuuksien ohella.

### 2.1 CSC yrityksenä

CSC on opetus- ja kulttuuriministeriön alainen voittoa tavoittelematon osakeyhtiö. CSC tarjoaa suurteholaskennan, tiedonsiirtokapasiteetin, tieteellisen laskennan sovellusten ja tiedon tallennuksen palveluita asiakkailleen. Asiakaskunta koostuu etupäässä yliopistoista, ammattikorkeakoulusta, tutkimuslaitoksista sekä museoista, kirjastoista ja arkistoista. (CSC 2014a.)

CSC on perustettu vuonna 1971 hallinnoimaan Suomeen hankittua Univac 1108 -suurtietokonetta. Koneen oli tarkoitus vastata Suomen yliopistoille tarvittavaan tiedonkäsittelytarpeeseen seuraavien 30 vuoden ajan. Luonnollisesti näin ei käynyt, vaan kone vanheni muutaman vuoden kuluessa. 1980-luvun lopussa Suomeen hankittiin ensimmäinen yhdysvaltalaisen Cray Inc. -yhtiön valmistama Cray XMP -supertietokone. Näihin aikoihin CSC:stä muodostui asiantuntijaorganisaatio vastaamaan supertietokoneen ja aiemmin avatun yliopistojen välisen Funet-tietoverkon käytön tuesta. CSC on hoitanut myös Suomen yhteyksiä pohjoismaiseen yliopistoverkkoon Nordunetiin. Tietoverkkojen nopeuden kehityttyä huimasti viimeisen parinkymmenen vuoden aikana on suomalaisille tutkijoille avautunut mahdollisuus käyttää CSC:n omien superkoneiden lisäksi myös Euroopan muuta tieteellistä suurlaskentakapasiteettia. Itse tietoteknisen laitteiston lisäksi CSC on tarjonnut alusta lähtien tutkijoille tieteellistä asiantuntemusta sekä laajan valikoiman tutkimuksissa käytettäviä ohjelmistoja ja tietokantoja. (CSC 2014b.)

Omien tieteelliseen tutkimukseen tuottamien palveluidensa lisäksi CSC toimii myös tutkijoiden siltana muihin eurooppalaisiin tiede- ja tutkimusorganisaatioihin ja niiden tietoteknisiin resursseihin. CSC:n panos korkeatasoisen suomalaisen tutkimusjärjestelmän kehittämiseen

kumpuaa kestävän kehityksen edelläkävijyyden visiosta sekä intohimoisen asiantuntijuuden ja ihmislähtöisyyden arvoista. (CSC 2014c.)

## 2.2 Kajaani DataCenter

CSC:n Kajaanin konesali on perustettu vuonna 2012 Renforsin Rannan yritysalueelle vanhan UPM-Kymmenen paperitehtaan varastohalliin (CSC 2014d). Kajaanin konesalissa toimii kolme supertietokonetta, joita käytetään suomalaisen tieteen ja tutkimuksen tekoon. Lisäksi Kajaanin konesalista tarjotaan tallennus- ja pilvipalveluita. Kajaanin pohjoinen sijainti luo ihanteelliset olosuhteet konesalin ja supertietokoneiden käyttöön. Viileä ilmasto tuo säästöjä jäähdytykseen, ja tämä näkyy pienentyneen energiankulutuksen myötä kustannustehokkuutena. (Nuotio 2014.)

Renforsin Rannan alueen infrastruktuuri on paperitehtaan jäljiltä hyvin luotettava. Konesalien tarvitsema sähkö ja tietoliikenne on moninkertaisesti varmistettu. CSC:n hallinnoima Funet-tietoverkko yhdistää Kajaanin konesalin CSC:n Espoon toimipisteeseen ja tutkimusjärjestöjen verkkoihin sekä Oulun että Kuopion kautta. Tietoliikenneyhteydet Funet-verkkoon näkyvät kuvasta 1. Renforsin Rannassa on ollut sähkökatko viimeksi vuonna 1982. (Rönty 2014.)



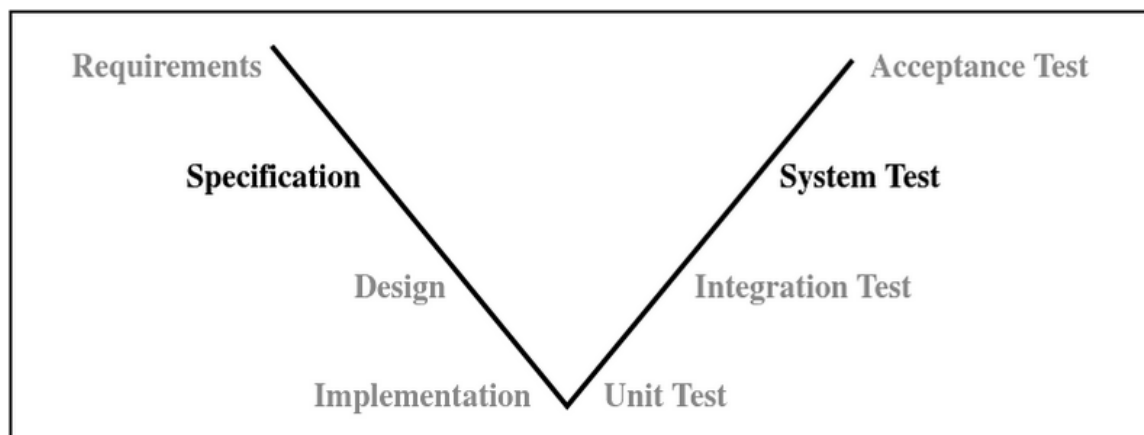
Kuva 1. Funet-verkkokartta (Funet-yhteydet)

### 2.3 Testiympäristön tarve CSC:llä

Aihe opinnäytetyöhön löytyi CSC:n todellisesta tarpeesta saada testausympäristö palvelemaan Kajaanin henkilöstöä, sekä osaltaan myös Espoon toimiston työntekijöitä heidän Kajaanin IT-palveluita koskevissa käyttötarpeissaan. Tämä projekti sopi hyvin Datacenter asiantuntija-koulutukseen, joten opinnäytetyön tekijälle annettu projekti suunnitella ja toteuttaa Kajaanin testausympäristö oli luonteva valinta koulutukseen liittyväksi opinnäytetyöksi. (Heiskanen 2014.)

### 3 TESTAAMINEN

Uudet ohjelmistot ja järjestelmät on testattava ennen käyttöönottoa. Testaamisella pyritään selvittämään kohteen toiminnan oikeellisuutta ja virheettömyyttä. Testaaminen etenkin sovelusten osalta tapahtuu yleensä monella tasolla niin sanotun V-mallin (kuva 2) mukaisesti. V-mallin mukainen testaaminen on jaettu moduuli- tai yksikkötestaukseen, integraatiotestaukseen, järjestelmätestaukseen ja hyväksymistestaukseen. Moduulitestauksessa pyritään löytämään yksittäisten ohjelmamoduulien virheitä. Integrointitestauksessa testataan eri moduulien välistä yhteistoimintaa. Järjestelmätestissä testataan lopulta koko sovelluksen tai järjestelmän toimivuutta ja suorituskkyä. Hyväksyntätestaus tehdään lopuksi ennen sovelluksen tai järjestelmän luovutusta tilaajalle tai loppukäyttäjälle. Tehtäessä järjestelmän lopullista järjestelmätestiä verrataan ohjelman tai järjestelmän toimintaa ja suorituskkyä toiminnalliseen määrittelyyn. Ohjelmistojen tapauksessa testaaminen ei saa perustua kirjoitettuun ohjelmakoodiin, vaan nimenomaan määrittely- ja suunnitteluvaiheessa tehtyihin vaatimuksiin ja määritelmiin. (Pohjonen 2002, 35–36.)



Kuva 2. V-malli (Watkins, J. 2001, 59)

Ohjelmistojen moduuleita testattaessa käytetään kahta toisiaan täydentävää lähestymistapaa. Sisäinen testaus testaa ohjelmistokomponentin sisäisten algoritmien virheettömyyttä. Tämä toteutetaan käymällä läpi sovelluksen sisäisiä tietovirtoja ja kontrollireittejä. Ulkoinen testaus ei ota kantaa ohjelman sisäisiin algoritmeihin, vaan sillä vertaillaan ohjelman tulosteita sille syötettyjen tietojen perusteella. Ulkoista testausta käytettäessä voidaan löytää ohjelmiston virheitä, mutta ulkoinen testaus ei paljasta, mistä virheet johtuvat. Tämän vuoksi ulkoista testaamista on täydennettävä sisäisellä testaamisella, jossa ohjelman sisäisistä algoritmeista löydetään

virheellisen tulosteen aiheuttamat virheet. Molempia testaustapoja käytettäessä tulee testausaineistoksi valita monipuolisia syöttöarvoja, joista osa on oikeanlaisia ja osa väärä. (Pohjonen 2002, 36.)

Muita testausstrategioita moduulitestien lisäksi on enemmän tämän työn tarvekartoitusta vastaava palautumistestaus, jossa testataan, kuinka sovellus tai järjestelmä palautuu poikkeustilasta tai virheestä. Suorituskykytestissä testataan, kuinka hyvin sovellus tai järjestelmä vastaa suorituskykyvaatimuksia. Läpikäynnit ovat testityyppejä, joissa ulkopuoliset asiantuntijat käyvät läpi järjestelmän vastaavuutta määrittelyyn tai kuvauksiin. Tarkastukset ovat läpikäyntien lailla ulkopuolisen toimijan suorittamia testejä. Tarkastukset ovat hyvin tarkkaan määriteltyjä testauksia, ja ne ovat verrattavissa auton katsastukseen. Täydellisen testauksen pitäisi paljastaa kaikki ohjelmiston virheet, mutta käytännössä tämä on mahdotonta. Vähänkin laajemmassa ohjelmassa erilaisia syötevaihtoehtoja ja ohjelman sisäisiä kontrollivirtoja on niin paljon, ettei täydellistä testausta voida suorittaa. Korkeintaan voidaan todistaa ohjelmassa olevan virheiden olemassaolo, ei ohjelman virheettömyyttä. (Pohjonen 2002, 36.)

### 3.1 Testaamisen historiaa

Tietokoneohjelmat ovat käyneet läpi testejä käytännössä niin kauan kuin tietokoneohjelmia on ollut olemassa. Sovelluskehityksen alkuaikoina testaaminen ei ole ollut kovinkaan tarkkaa ja muodollista, vaan ohjelmien virheiden etsintä on ollut osa ohjelmistokehitystä. Ohjelmistotuotannon kehittyessä ja ohjelmistojen laajentuessa pienistä yhden käyttäjän yksinkertaisista ohjelmista laajoiksi usean käyttäjän ohjelmistoiksi ja tietoverkkojen kehittyessä verkkojen yli käytettäviksi on testaamisen ollut pakko kehittyä mukana. Testaamisesta onkin tullut yksi tärkeä erikoistumiskohde ohjelmistokehityksessä. Testitavat ja tekniikat ovat kehittyneet testauksen ammattilaisten myötä. (Watkins, J. 2001, 8–9.)

### 3.2 Testaamisen lähtökohdat

Alun perin testaaminen määriteltiin toiminnaksi, jolla pyrittiin varmistamaan ohjelman tai järjestelmän ominaisuuden tai kyvyn toimia vaatimustensa mukaisesti. Tätä lähestymistapaa kutsutaan testauksen perinteiseksi lähestymistavaksi tai *positiiviseksi testaamiseksi*. Toinen näkökulma testaamiseen on, että ohjelmaa tai järjestelmää ajetaan tarkoituksena löytää sen virheet.

Tämä näkökulma ei ole yhtä havainnollinen, eikä se huomioi ohjelman vastaavuutta vaatimuksiinsa. Tämä näkökulma auttaa testaamaan ohjelmaa minkä tahansa vaatimusmäärittelyn ulkopuolisen virheen varalta. Tätä tapaa kutsutaan *negatiiviseksi testaamiseksi*. Käytännössä testattaessa käytetään sekä positiivista että negatiivista lähestymistapaa. (Watkins, J. 2001, 9.)

Testaaminen voidaan määritellä myös riskinhallinnan näkökulmasta. Testaaminen nähdään prosessina jolla pyritään tutkimaan ja ymmärtämään ohjelman hyötyjen ja riskien tilaa. Tämän määritelmän myötä testaajan vastuuna on hallita ja torjua sovelluksen tai järjestelmän vikaantumisia ja vikaantumisen aiheuttamia vahinkoja käyttäjille. Testaaja joutuu tältä näkökannalta testattaessa käyttämään muiden testitoimien lisäksi myös riskinhallinnan strategioita. Riskit ja niiden hallinta ovat sovelluskehittäjille tuttuja projektinhallinnan osalta, ja siksi projektinhallinnassa käytettyjä työkaluja ja tekniikoita voidaankin käyttää hyödyksi myös testauksessa. (Watkins, J. 2001, 9–10.)

### 3.3 Järjestelmätestaus

Järjestelmätestaus on testauksen osa, missä yhteen liitettyjen ohjelmakomponenttien toimivuutta testataan kokonaisuutena. Järjestelmätestaus suoritetaan integraatiotestausten valmistuttua. Järjestelmätestausta seuraava vaihe on hyväksymistestaus. Järjestelmätestauksessa järjestelmää testataan V-mallin mukaisesti toiminnallisessa määrittelyssä esiteltyjä vaatimuksia vastaan. Järjestelmätestauksessa käytetään ulkoisen testaamisen menetelmiä ja testikohteena ovat nimenomaan määrittelyvaiheessa tehdyt korkean tason vaatimukset. Järjestelmätestauksessa ei oteta kantaa järjestelmän komponenttien tekniseen toteutukseen vaan nimenomaan järjestelmän syötteiden tuomaan oikeanlaiseen tulosteeseen. Testattavat ominaisuudet ja testiproseduurit on luotava teknistä määritelmää tehtäessä. Teknisestä määritelmästä ja järjestelmäsuunnittelun vaiheista kerrotaan luvussa 4.1. (Watkins, J. 2001, 59–60.)

Järjestelmätestaus tulee toteuttaa seuraavien osien kautta:

- Vertaa järjestelmää teknisen määrittelyn dokumenttiin ja pyri todentamaan seuraavia asioiden vastaavuus: järjestelmän korkean tason toiminnallisuudet, toimintaympäristön tiedonkäsittelyn ja transaktiotasot, järjestelmän toimintavaatimukset, varmistus- ja palautusvaatimukset sekä turvallisuusvaatimukset.
- Tarkista järjestelmän vaatimukset ulkoisten järjestelmien kanssa kommunikointiin.



- Vertaa järjestelmän toimivuusvaatimuksia tulevaan käyttöympäristöön ja pyri löytämään mahdolliset ristiriidat.
- Tutki asennus-, varmistus-, palautusprosessien ohjeistukset ja ohjeiden saatavuudet.
- Käytä yllä mainittuja tietoja testitapausten luomiseen ja tee niistä tarvittavat testiprosessit.

Järjestelmätestaamisessa voi käyttää aiemmissa testivaiheissa, yksikkötestaus ja integrointitestaus, käytettyjä testitapauksia. (Watkins, J. 2001, 60.)

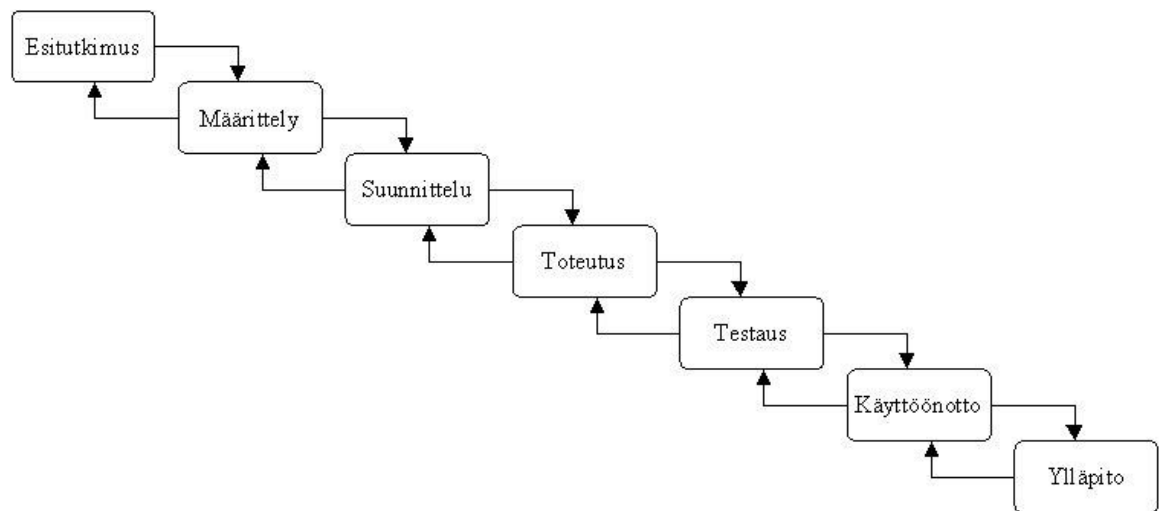
Koska järjestelmätestauksen tarkoitus on myös valmistautua hyväksyntätestaukseen, on tärkeää, että testiaineisto on tarkkaa ja mahdollisuuksien mukaan todellista aineistoa, jota järjestelmään tullaan syöttämään. Testiaineisto voi olla kopioitu vastaavasta aiemmasta järjestelmästä, mikäli uutta reaaliaikaista aineistoa ei voida syöttää esimerkiksi toimintajärjestelmän toiminnan vaarantumisen tai tietoturvariskien vuoksi. Mikäli testiaineistoa joudutaan muuttamaan tietoturvasyistä, tulee testattaessa olla erityisen huolellinen, ettei aineiston eheys kärsi ja aiheuta testitulosten vääristymistä. (Watkins, J. 2001, 60–61.)

## 4 MÄÄRITTELY JA SUUNNITTELU

Opinnäytetyön käytännön osuudessa käytettiin apuna ohjelmisto- ja tietojärjestelmäkehityksessä yleisesti käytettyjä menetelmiä. Tarkkaa projektikuvausta asiaan kuuluvine dokumentteineen ei tehty, mutta työ tehtiin kuitenkin perinteisen, kuvassa 3 näkyvän vesiputousmallin mukaisesti. Tässä luvussa kuvataan tietojärjestelmäprojektin vesiputousmalli teoriatasolla ja sen jälkeen kuvataan työn kunkin vaiheen aikana tehdyt toimenpiteet.

### 4.1 Vesiputousmalli

Tietojärjestelmän kehittämisen alkuun panevana tarpeena on täyttää jokin käyttötarve, mitä olemassa olevilla laitteistoilla ei voida toteuttaa. Kehitystyö on systemaattinen ja usein projektinomainen sarja vakiintuneita vaiheita. Näiden vaiheiden lopputulos on toimiva, testattu ja dokumentoitu järjestelmä. Sovellusten kehittämisessä sekä sovellusten ja laitteistojen muodostamien laajempien järjestelmien kehittämisessä on käytössä erilaisia elinkaarimalleja. Yksi jo 1960-luvun lopussa kehitetty elinkaarimalli, joka on edelleen yksi käytetyimmistä malleista, on vesiputousmalli. Siinä suoritetaan aina yksi vaihe kerrallaan, jonka valmistuttua siirrytään seuraavaan. Vesiputousmallin vaiheet näkyvät kuvassa 3. Vesiputousmallissa toteutuu etenkin sovelluskehityksessä olennainen iteratiivisuus varsin huonosti, mutta siitä huolimatta se on edelleen yleisessä käytössä. Iteratiivisuus tarkoittaa tietynlaisen vaiheen aikana havaittujen virheiden korjaamista palaamalla suorittamaan edellinen vaihe uudestaan. (Hiltunen 2014.)



Kuva 3. Vesiputousmalli (Hiltunen 2014)

Vesiputousmallissa järjestelmän kehitystyö alkaa esitutkimusvaiheella, jossa tuotetaan tietoa tarpeesta kehittää uusi järjestelmä. Esitutkimuksen tuottama raportti pitää sisällään tietoa ja lähtökohtia yrityksen päättäjien tarpeeseen. Päättäjät tekevät päätöksen kehitystyön käynnistämisestä. Esitutkimusraportissa kuvataan nykyinen kehitystä kaipaava tilanne ja nykytilanteen ongelmat. Raportti esittelee viite- ja sidosryhmät, joita kehitettävä järjestelmä tulee koskettamaan jollakin tavalla. Esitutkimus määrittää kehitettävän järjestelmän alustavat tavoitteet ja järjestelmää koskevat rajoitteet. Hankkeen läpiviemisen suunnitteleminen ja eri toimintavaihtoehtojen läpikäyminen kuuluvat myös esitutkimusvaiheeseen. (Hiltunen 2014.)

Toinen vaihe kehitystyössä on määrittely. Määrittelyvaiheen lopputuote on vaatimusmäärittelydokumentti. Dokumenttiin kerätään kaikilta järjestelmään liittyviltä sidosryhmiltä heidän vaatimuksensa järjestelmän tarpeista. Määrittelyvaiheessa ei oteta kantaa järjestelmän tekniseen toteutukseen, vaan ainoastaan listataan toimintoja ja ominaisuuksia, joita järjestelmän tulee täyttää. Nämä vaatimukset toiminnoista ja ominaisuuksista jaetaan toiminnallisiin ja ei-toiminnallisiin vaatimuksiin. Toiminnallisiin vaatimuksiin kuuluvat tiedot siitä, mitä järjestelmän odotetaan tekevän. Toiminnallisiin vaatimuksiin kuuluvat järjestelmän kommunikointitavat ympäristönsä kanssa ja määritelmä kuinka käyttäjät työskentelevät järjestelmän kanssa. Ei-toiminnallisiin vaatimuksiin kuuluvat järjestelmän rajoitteet, kuten vasteajat ja käytettävyyteen liittyvät seikat. Ei-toiminnallisia vaatimuksia voivat olla esimerkiksi, kuinka nopeasti järjestelmän on vastattava pyyntöihin verkon yli tai millä käyttöjärjestelmällä järjestelmää voidaan käyttää. Rajoitteina voi olla esimerkiksi yhtäaikaisten käyttäjien määrä tai tilausjärjestelmän osalta yhden käyttäjän tekemien yhtäaikaisten tilausten määrä. (Hiltunen 2014.)

Vaatimukset sidosryhmiltä kerätään käyttäen haastatteluja, aivoriihiä ja palavereita. Vaatimuksia saadaan myös analysoimalla vanhoja tietojärjestelmiä ja aiheesta tehtyjä markkinatutkimuksia. Vaatimuksiin vaikuttavat olennaisesti myös ulkoiset tekijät, kuten lainsäädäntö, standardit ja asetukset. Ongelmia vaatimusmäärittelyä tehtäessä voi tulla keskeneräisten vaatimusten vuoksi ja eri ryhmien vaatimusten välisten ristiriitojen takia. Ongelmien ja epäselvyyksien välttämiseksi vaatimukset tulee kirjata mahdollisimman tarkasti. Vaatimusmäärittelydokumentissa tulisi käydä ilmi kuvaus kehittämishankkeen toimeksiannosta, ja siinä tulisi kuvata yleisellä tasolla kohdejärjestelmän nykyinen tila. Dokumentissa kuvataan järjestelmälle asetetut tavoitteet, rajoitteet ja lisäselvitykset. Kaikki toiminnalliset ja ei-toiminnalliset vaatimukset, kuten myös rajoitteet, kuvataan tarkasti ja priorisoidaan selkeästi. Kaikki kohdat numeroidaan, jotta kyseiseen kohtaan voi viitata helposti. (Hiltunen 2014.)

Määrittelyvaiheen vaatimusmäärittelydokumentista johdetaan seuraavaksi dokumentiksi toiminnallinen määrittely, jossa selvitetään, mitä järjestelmän tulee tehdä. Toiminnallisessa määrittelyssä analysoidaan edellisessä vaiheessa tehty vaatimusmäärittely ja otetaan kantaa yksityiskohtaisemmin, kuinka eri toiminnot tulevat toimimaan. Kaikki toiminnot kuvataan yksityiskohtaisesti. Toiminnalliseen määrittelyyn kuuluvat myös järjestelmän käsittelemien tietojen, tietokantojen ja rajapintojen tarkat kuvaukset. Toiminnallisen määrittelyn tuottaman dokumentin on oltava erittäin selkeä, jotta suunnitteluvaiheessa ei tule vääriä tulkintoja. (Hiltunen 2014.)

Määrittelyvaiheessa otettiin kantaa järjestelmään yleisellä tasolla ilman tekniseen toteutukseen pohdintaa. Seuraavana vaiheena järjestelmän kehityksessä on suunnittelu, jolloin järjestelmän teknistä toteutusta aletaan suunnittelemaan aiemman vaiheen tuottamien määrittelydokumenttien vaatimusten toteuttamiseksi. Suunnitteluvaihe muuttaa toiminnallisen määrittelyn teknisiksi määrittelyiksi. Teknisen määrittelyn dokumentti kertoo, kuinka järjestelmä toteutetaan. Dokumentissa on kuvaukset järjestelmän laitteisto- ja ohjelmistoympäristöstä, ohjelmisto- ja tietokanta-arkkitehtuurista, järjestelmän jokaisesta moduulista ja alijärjestelmästä. Dokumenttiin kuvataan myös mahdolliset vaihtoehtoiset ja hylätyt ratkaisut sekä toteutuksen reunaehdot. (Hiltunen 2014.)

Suunnitteluvaiheen jälkeen neljäntenä vaiheena on toteutus. Toteutusvaiheessa rakennetaan aiempien vaiheiden myötä saadut vaatimukset toteuttava järjestelmä käyttäen ennalta valittuja menetelmiä. Järjestelmän tulee toteuttaa kaikki aiemmissä määrittelyissä tehdyt vaatimukset ja sen tulee vastata suunnitteluvaiheessa tehtyjä teknisiä suunnitelmia. Toteutusvaiheen

loppupuolella järjestelmän sovellukset ja laitteet integroidaan toimivaksi kokonaisuudeksi. Toteutusvaiheeseen kuuluu olennaisesti järjestelmän laatu. Laatuun liittyviä seikkoja on järjestelmän toiminnallisuus, luotettavuus, siirrettävyys ja ylläpidettävyys. (Hiltunen 2014.)

Kun järjestelmä on toteutettu, sitä ei oteta vielä käyttöön, vaan ennen käyttöönottoa järjestelmän toimivuus tulee testata. Testausvaiheessa järjestelmää verrataan toiminnallisen määrittelyn vaatimuksiin. Virheitä etsitään järjestelmän suorituskyvystä ja toiminnoista. Yksittäisiä järjestelmän osia testataan, jotta niiden sisäinen toimivuus saadaan selville. Yksittäisten osien toimivuuden testauksen jälkeen testataan osien yhteen toimivuus integroititesteillä. (Hiltunen 2014.)

Onnistuneen testauksen jälkeen järjestelmä voidaan lopulta ottaa käyttöön. Mikäli uusi järjestelmä korvaa jonkin aiemmin olemassa olleen järjestelmän, tulee käyttöönottovaiheessa siirtää kaikki vanhan järjestelmän tiedot uuteen järjestelmään. Vanhaa ja uutta järjestelmää ajetaan usein rinnakkain jonkin aikaa. Iso osa uuden järjestelmän käyttöönotossa on käyttäjien kouluttaminen ja käyttöohjeistusten luominen. (Hiltunen 2014.)

Viimeinen vaihe hankkeessa on ylläpitovaihe. Tämä on hankkeen pisin yksittäinen vaihe. Ylläpitovaiheeseen kuuluvat järjestelmän vikojen selvitys ja korjaaminen sekä olemassa olevien toimintojen muutokset. Ylläpitoon kuuluvat myös jatkokehityshankkeet ja uusien toiminnallisuuksien suunnittelu ja toteutus. Ylläpitovaiheen kannalta on erittäin tärkeää, että järjestelmän kehitysvaiheessa on luotu tarkka ja kattava dokumentaatio. (Hiltunen 2014.)

## 4.2 Testausympäristön teoriavaiheet

Tässä luvussa käsitellään opinnäytetyön aiheena olevan järjestelmän toteutuksen teoriaa. Luvussa käsitellään vesiputousmallin vaiheet esitutkimuksesta määrittelyn kautta suunnitteluun. Loput vaiheet, käytännön toteutuksesta eteenpäin, käydään läpi luvussa 5.

### 4.2.1 Esitutkimusvaihe

Kajaanin henkilökunta on keskustellut jo jonkin aikaa tarpeesta saada oma testausympäristö uusien sovellusten ja asetusten testaamiseen. Syksyllä 2014 viikkopalaverissa päätettiin alkaa viemään asiaa ideasta käytäntöön ja opinnäytetyön tekijä sai tehtäväkseen alkaa määritellä ja

suunnitella järjestelmää. Palaverissa käytiin läpi lähtökohtia ja järjestelmän pääasiallisia käyttötarpeita. Järjestelmään päätettiin ottaa mallia Espoossa olemassa olevasta järjestelmästä. Kajaanin oman testausjärjestelmän hankinta on perusteltua, sillä Kajaanin Konesalin sisäisiä verkkoja ja sen suorituskkyä voi helpoiten testata Kajaanissa sijaitsevalla järjestelmällä. (Heiskanen 2014.)

Espoossa olemassa oleva testiympäristö koostuu yhteensä neljästä palvelimesta. Itse virtualisointialustan isäntäkoneen roolissa on kolme Dellin PowerEdge R710 -palvelinta, joissa jokaisessa on kaksi kappaletta Intel Xeon E5540 -prosessoria ja 192 gigatavua keskusmuistia (RAM). Neljäs kone on vanhempi Dellin PowerEdge 1850, jossa on 4 gigatavua keskusmuistia ja 3 GHz kellotaajuudella toimiva Intel Xeon -neliydinprosessori. Tämä palvelin on pelkästään VMware vCenterin käytössä. Järjestelmä on kytketty tietoverkkoon 1 Gb/s *Ethernet*-kaapeloinnilla. Tallennusverkko käyttää 8 Gb/s *Fiber channel* -valokuituverkkoa. Testiympäristöllä ei ole omaa tallennusjärjestelmää, vaan sen käytössä on kahdesta eri tuotantokäytössä olevasta levyjärjestelmästä lohkaistut osat. (Niemi 2014.)

Espoon testiympäristöä käytetään pääasiallisesti palvelukehityksen testialustana ja uusien VMwaren ohjelmistoversioiden testaamiseen ennen tuotantokäyttöön ottoa. Käyttäjinä ovat koko henkilöstö CSC:n ICT-palveluita tuottavasta organisaatioyksiköstä ja CSC:n ohjelmistokehitystiimit. Käyttäjien saamien resurssien määrää ei ole rajoitettu, vaan jokaiselle luotavalle virtuaalikoneelle annetaan resursseja tarpeen mukaan. Testiympäristöä valvotaan Dellin Foglight for Virtualization -monitorointityökalulla. Sovelluspuolella järjestelmä koostuu VMware ESXi 5.5 -palvelinsovelluksista ja klusteriksi ESXi-palvelimet muodostetaan käyttäen VMware vCenter Server -sovellusta. (Niemi 2014.)

#### 4.2.2 Määrittely

Testausympäristön käyttötarkoituksesta varten on haastateltu CSC:n Kajaanin toimipisteen henkilökuntaa, ja näiden haastattelujen perusteella on listattu erilaisia järjestelmän käyttökohteita. Käyttökohteiden avulla on voitu luoda vaatimusmäärittely järjestelmältä vaadittavien ominaisuuksien kartoittamiseen. Tämän kartoituksen perusteella on voitu tehdä päätös tarvittavien laitteistojen valinnasta. Haastattelut on suoritettu syksyn 2014 aikana ennen laitehankintoja. Käyttötarpeet keskittyvät eri käyttöjärjestelmäversioiden, hallintaohjelmien ja tietotur-

vallisuuden testaamiseen. Verkkojen ja palomuurien testaaminen onnistuu sisäisten virtuaaliverkkojen lisäksi käyttäen Kajaanissa olevaa testiverkkoa. Eräs tärkeä testattava asia on järjestelmän ja sovellusten vikasietoisuuden testaaminen erilaisilla asetuksilla: osaako järjestelmä toipua automaattisesti yhden isäntäkoneen vikaantuessa tai sammuesssa ja miten sovellukset käyttäytyvät, jos tietoverkko katkeaa ja liikenne siirtyy automaattisesti varayhteydelle. Järjestelmällä halutaan testata myös viasta toipumisen viiveitä erilaisilla asetusvaihtoehdoilla.

Testiympäristön avulla tehdyillä vikasimulaatioilla ja parhaiden asetusmäärittelyiden selvittämisellä saadaan käytännössä testattua tietoa uusien järjestelmien dokumentointiin ja vikatilanteista toipumisen ohjeistuksiin.

### Käyttökohteet

Järjestelmän käyttötarkoituksena kerättiin järjestelmän tulevilta käyttäjiltä esimerkkejä heidän käyttötarpeistaan. Käyttötarpeet keskittyivät lähinnä verkkojen, ohjelmistojen vikasietoisuuden ja käyttöjärjestelmä- ja ohjelmistoversioiden vertailuun ja testailuun.

- Erilaisten korkean käytettävyyden ja vikatilanteesta selviämisen vuoksi käytettyjen kahdenkannusten testaaminen
- Tietoverkon layer2 ja layer3 katkojen sovelluksiin aiheuttaman vaikutuksen selvittäminen ja kuinka katkoista toivutaan
- Oikeiden tietoverkkoasetusten löytäminen ja dokumentoiminen erilaisille asetuskoonpanoille
- Verkkolaiterikkojen simulointi ja niiden järjestelmiin aiheuttamien vaikutusten tutkiminen
- Verkon ulkopuolisten IP-yhteyksien testaaminen käyttäen erillistä testiverkkoa
- Palomuuriympäristön verkkoyhteyksien perustestaaminen
- Palomuurisääntöjen testaaminen
- Palomuurien vaikutus tietoliikenteen suorituskykyyn

(Karppinen 2014a)

- Käyttöjärjestelmien eri versioiden testaaminen ja beta/release-versioiden vertailu
- Hallintaohjelmistojen testaaminen eri käyttöjärjestelmillä ja eri käyttöjärjestelmäversioilla
- Uusien palveluohjelmistojen ja -alustojen testaus

(Niininen 2014a)

- Kajaanin ja Espoon konesalien välisen vMotion-virtuaalikoneiden siirtoyhteyden testaaminen

(Heiskanen 2014)

#### 4.2.3 Suunnittelu

Tässä luvussa kerrotaan, kuinka järjestelmää lähdettiin suunnittelemaan määrittelyvaiheessa tehdyn käyttötapauskartoituksen myötä saatujen vaatimusten toteuttamiseksi. Luvussa kuvataan laitteiston ja sovellusten valintaa, verkko- ja tallennusratkaisujen teknologiaa ja kuinka ne suunniteltiin otettavaksi osaksi ympäristöä. Lisäksi luvussa käydään läpi järjestelmän asetuksia ja sovellusten laitevaatimuksia käyttäen ohjeena VMwaren ”best practises” -käytäntöjä ja asennusoppaita.

##### Laitteisto

Määrittelyvaiheessa kerättyjen käyttötarpeiden myötä kävi ilmi, että vaatimukset täyttääkseen järjestelmässä on oltava vähintään kaksi fyysistä palvelinta. Tämä on välttämättömyys, jotta vaadittuja kahdennuksia ja vikatilanteesta toipumisen testejä voitaisiin suorittaa. Kahta palvelinta käytettäessä voidaan simuloida järjestelmän kaatumista sammuttamalla toinen kone, jolloin virtuaaliympäristön oikein määriteltyjen vikasetteijäasetusten myötä virtuaalikoneet ja niillä ajettavat palvelut pitäisi käynnistyä automaattisesti toisella palvelimella.



Laitteistoa valittaessa suurin painoarvo oli yrityksessä aiemmin käytetyillä laitemerkeillä ja mallilla sekä laitteiden hankintahinnalla. Laitteiden hankinnassa vaihtoehtoina on hankkia kokonaan uudet laitteet ja asentaa järjestelmä niin sanotusti puhtaalta pöydältä. Toinen, halvempi vaihtoehto on käyttää jo olemassa olevia laitteistoja. Uusien laitteiden valinnan suhteen päättävältä laitemerkin ja mallin osalta ei yrityksen IT-politiikan vuoksi ollut opinnäytetyön tekijällä, vaan eri osastolla, joka laitehankinnoista vastaa. Kyseisen osaston päätöksellä päädyttiin loppujen lopuksi käyttämään olemassa olevia laitteita. Kyseiset palvelinlaitteet ovat olleet osa isompaa virtualisointiympäristöä, joten niiden laitteistovaatimukset olivat ennalta testatut suunnitellulle virtualisointiratkaisulle.

Molemmat palvelinkoneet ovat Dell PowerEdge R520 -mallia. Kummassakin palvelimessa on yksi 2,2 GHz kellotaajuudella toimiva Intel Xeon E5-2407 -prosessori ja 48 gigatavua keskusmuistia. Kummankin koneen paikallisena tallennusratkaisuna on kaksi 300 GB kiintolevyä, joiden vikasietoisuutta on parannettu ottamalla käyttöön RAID 1, eli peilaus. Verkkoyhteyksiä varten palvelimissa on kaksi 1 Gb/s Ethernet-verkkoliitäntää ja yhteensä kuusi 10 Gb/s valokuituliitäntää. Virransaannin varmistamiseksi palvelimissa on kaksi virtalähdettä, joista toinen on kytketty UPS:iin.

### Tallennusjärjestelmä

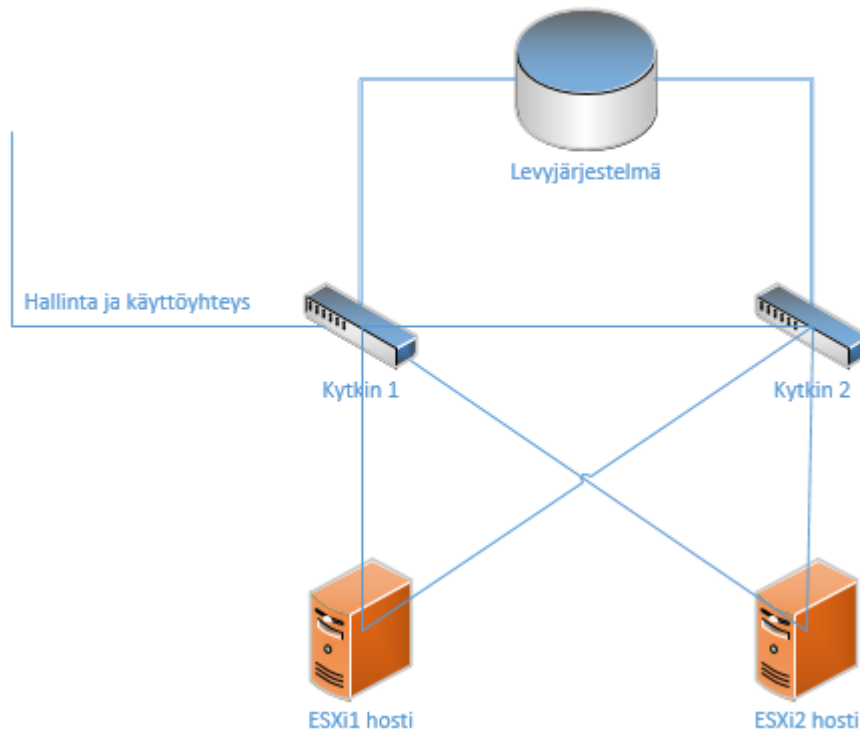
Levyjärjestelmän osalta päädyttiin samasta syystä kuin palvelinten tapauksessa, käyttämään uuden vain tätä järjestelmää palvelevan levyjärjestelmän sijaan osaa olemassa olevasta isommasta levyjärjestelmästä. Levytilaa varattiin järjestelmästä yhteensä 8 TB, joka jaettiin kolmeksi eri loogiseksi levyalueeksi eli LUN:iksi. LUN:it varattiin eri levytyypeiltä niiden käyttötarkoituksen mukaan. Käyttöjärjestelmien asennustiedostoja eli levykuvia (*image*) varten luotiin 2 TB RAID 6 -määritettyä levytilaa. Käyttöjärjestelmien asennusta ja käyttöä varten luotiin niin ikään 2 TB levytilaa, mutta tämä tila varattiin nopeammalta RAID 50 -määritetyltä ja nopeampia levyjä sisältävältä levyalueelta. Kolmanneksi levyalueeksi yleiseksi tallennustilaksi luotiin 4 TB kokoinen alue hitaammalta RAID 6 -levyalueelta. (Niininen 2014b.)

Tallennusjärjestelmien yhteydessä käytetään usein termiä RAID. Se on lyhenne sanoista Redundant Array of Inexpensive Disks. Tämä tarkoittaa kahden tai useamman kiintolevyn muodostamaa kokonaisuutta, jossa käytössä on jokin nopeutta, vikasietoisuutta tai molempia parantava määrittely. Levyjärjestelmään saadaan nopeutta käyttämällä hajautettua tiedontallennusta. Tällöin levyjärjestelmä kirjoittaa datan osina usealle levyille yhtä aikaa. Vikasietoisuuden

lisäämiseksi tieto voidaan peilata toiselle levyille, tai kirjoitettavasta tiedosta voidaan laskea pariteettibitti, joka tallennetaan erilliselle kiintolevyille. Tällöin vikaantuneen kiintolevyn tieto voidaan palauttaa uudelle levyille käyttämällä pariteettia ja muiden kiintolevyjen tietoa. RAID-määrittäjiä kontrolloi levyjärjestelmässä erillinen RAID-ohjain. Tässä työssä käsitellään kolmea eri RAID-tasoa. RAID 1 on vikasietoisuutta parantava ominaisuus. Kiintolevyille tallennettu tieto kopioidaan sellaisenaan toiselle kiintolevyille. Mikäli levy hajoaa, ovat tiedot tallessa toisella levyllä. RAID 5 yhdistää nopeuden ja vikasietoisuuden jakamalla kirjoitetun tiedon useammalle levyryhmälle, joista jokaisen ryhmän tiedosta lasketaan pariteetti joka tallennetaan eri levyille kuin itse tieto. Ryhmien sisällä tieto jaetaan niin ikään eri levyille. Kokoonpano on nopea ja kestää yhden levyn hajoamisen ryhmää kohti ilman tiedon menetystä. RAID 6 -tasossa tieto jaetaan usealle levyille ja tiedosta lasketaan kaksi pariteettia eri levyille. Järjestelmä kestää kahden levyn hajoamisen ilman tiedon menetystä. (Lynn 2014.)

Tallennusjärjestelmä koostuu kahdesta Dell EqualLogic PS6100E -levyjärjestelmästä. Kummassakin niistä on 24 kiintolevyä. Toisessa levyjärjestelmässä on 24 kappaletta 2 TB kokoista 7200 kierrosta minuutissa pyörivää levyä. RAID 6 -määrittäksellä tallennustilaa on 33,74 TB. Toisessa levyjärjestelmässä on 24 kappaletta pienempiä 600 GB kokoista nopeaa 15000 kierrosta minuutissa pyörivää kiintolevyä. Nämä levyt on määritetty RAID 50 -tasolle, joten tallennustilaa tässä nopeammassa levyjärjestelmässä on 9,42 TB. Levyjärjestelmän levyosioihin pääsy on rajattu IP-osoitteen perusteella ja CHAP-autentikointiprotokollalla. (Niininen 2014b.)

Tallennusverkon muodostaa kahdennettu 10 Gb/s nopeudella toimiva valokuituyhteys, jonka fyysiset kaapeloinnit näkyvät kuvassa 4. Kummastakin palvelimesta lähtee yksi valokuitu ensisijaiselle runkokytkimelle ja toinen valokuitu toiselle runkokytkimelle. Kytkimistä menee niin ikään kahdennettu valokuituyhteys levyjärjestelmään. Levyjärjestelmän tietoliikenne kulkee iSCSI-protokollaa käyttäen. (Karppinen 2014b.)



Kuva 4. iSCSI ja vMotion-liikennettä kuljettava valokuituverkko

iSCSI on tietojärjestelmien levyjärjestelmäliikenteen käsittelyä varten luotu protokolla. iSCSI käyttää TCP-protokollaa iSCSI-komentojen siirtoon. Tämä mahdollistaa tallennusverkon rakentamisen olemassa olevan TCP/IP-lähiverkon päälle. iSCSI mahdollistaa näin ollen huomattavan kustannussäästön perinteiseen *Fiber Channel SAN*-verkkoon nähden. iSCSI koostuu kohteesta (*target*), joka on levyjärjestelmä, ja aloittajasta (*initiator*) eli palvelimesta, joka avaa yhteyden levyjärjestelmään. iSCSI-initiaattori voi olla ohjelmallinen, laitesidonnainen tai laitteistoriippumaton. VMwaren ohjelmallinen iSCSI-sovitin on sisäänrakennettu ESXi:n VMkerneliin. Sovitin mahdollistaa palvelimen iSCSI-liikenteen ajamisen tavallisten verkkokorttien läpi. iSCSI-initiaattorin ja kohteen välinen istunto identifioidaan *iSCSI session ID*:llä eli ISID:llä. ISID ei ole sidottu laitteistoon, vaan istunto säilyy, vaikka laitteistosta jouduttaisiin vaihtamaan verkkokortti. (VMware 2013a, 4–6.)

IP-verkot, joita iSCSI käyttää, ovat oletuksena salaamattomia. Tämän vuoksi iSCSI pitää sisälleen salausprotokollan, jolla tiedonsiirron tietoturvaa saadaan parannettua. Tämä salausprotokolla on nimeltään Challenge Handshake Authentication Protocol eli CHAP. CHAP käyttää kolmivaiheisen kättelyn algoritmia iSCSI-initiaattorin ja kohteen identiteetin varmistamiseen. Varmistus tapahtuu vertailemalla ennalta määriteltyä salaista avainta, jonka sekä initiaattori että

kohde tietävät. ESXi tukee CHAP autentikointia kohdekohtaisesti. Eri suojaustason levyaluilla voi olla eri CHAP-salasanat. ESXi tukee yksisuuntaista CHAP:ia kaikille iSCSI-initiaattorityypeille ja kaksisuuntaista CHAP:ia ohjelmalliselle initiaattorille ja osalle laitetason initiaattoreille. CHAP:ia käyttöönotettaessa on ensin tehtävä määrytykset levyjärjestelmän puolelle, jonka jälkeen ESXi:n asetuksissa tehdään vastaavat määrytykset. (VMware 2014b.)

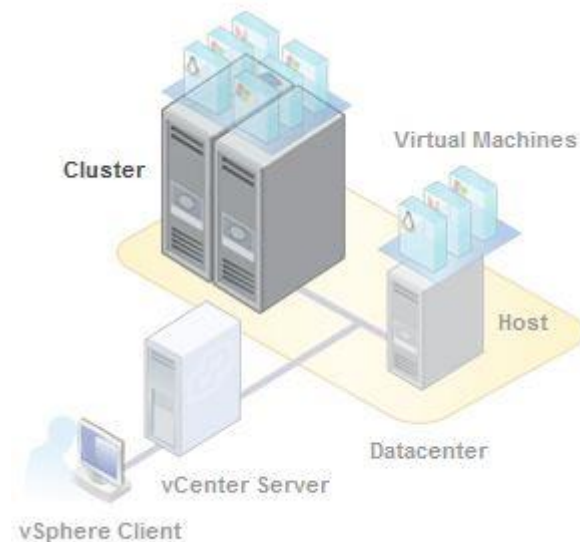
### Virtualisointiratkaisut

Virtualisointi on teknologia, jolla voidaan hämärtää tietokoneen fyysisen laitteiston ja ohjelmiston välinen yhteys. Virtualisointialusta asennetaan fyysiseen laitteeseen, jonka jälkeen virtualisointialustan päälle voidaan luoda virtuaalisia tietokoneita. Virtuaalisen tietokoneen näkökulmasta katsottuna niiden toimintaympäristö on normaali tietokonelaitteisto. Todellisuudessa virtuaalikoneet näkevät fyysisen laitteiston sijaan virtualisointialustalle luodun loogisen näkyvän laitteistosta. Virtualisoinnilla saadaan aikaan muutamia hyvin hyödyllisiä asioita: Virtualisoinnilla voidaan yhdistää usean fyysisen laitteiston resurssit yhdeksi resurssikokonaisuudeksi, niin sanotuksi resurssipooliksi. Toinen etu on usean virtuaalisen tietokoneen ajaminen yhdellä laitteistolla. Tällä saadaan parannettua laitteiston käyttöastetta ja hyötysuhdetta. Virtualisointi tuo järjestelmään mukanaan myös korkean käytettävyyden, vikatilanteesta toipumisen ja kuorman tasauksen ominaisuuksia. (Waters, J, 2007.)

Virtualisointiratkaisun tarjoajaksi valittiin toimeksiantajan aikaisempien virtualisointijärjestelmien tapaan markkinoiden luultavasti tunnetuin valmistaja, VMware. VMwaren käytöstä ja ylläpidosta on paljon kokemusta organisaation sisällä, joten se oli luonnollinen valinta. Järjestelmä on tarkoitus olla käyttäjille helposti käytettävä, ja näin ollen oli perusteltua käyttää kaikkien ennestään tuntemaa virtualisointiympäristöä Microsoftin Hyper-V:n tai Citrixin XenServerin sijaan. Lisäksi yrityksellä on valmiiksi lisenssi VMwaren tuotteilla toteutettuun järjestelmään. VMwaren tuotteiden ollessa kyseessä tarkoittaa se käytännössä sitä, että palvelimiin asennetaan VMware ESXi -virtualisointiohjelmisto. ESXi:stä valittiin uusin versio 5.5.

VMware Inc. on yhdysvaltalainen, vuonna 1998 perustettu, Kalifornian Palo Altossa sijaitseva ohjelmistoyritys. VMware tarjoaa monentasoisia työkaluja tietotekniikan virtualisointiin. Yksinkertaisin sovellus on Player, jolla voidaan ajaa virtuaalikoneita tavallisella työpöytätietokoneella. Workstation on monipuolisempi työkalu, joka on tarkoitettu ammattikäyttöön. VMware ESXi Server on yrityksille suunnattu virtualisointialusta, joka toimii suoraan palvelinlaitteeseen asennettuna ilman välissä olevaa käyttöjärjestelmää. VMware vCenter Server

-sovellus tarjoaa mahdollisuuden luoda useasta ESXi Serveristä klusteri ja hallita sitä keskitetysti. Kuva 5 havainnollistaa ESXi-palvelinten eli hostien muodostamaa klusteria ja sen hallintaa vCenter Serverillä. (Crunchbase.)



Kuva 5. VMware-ympäristö (Doxer 2014)

Jotta virtuaaliympäristön isäntäkoneet saataisiin toimimaan yhdessä ja jotta järjestelmään voitaisiin ottaa mukaan kuorman tasausta ja vikasietoisuutta mahdollistavia ominaisuuksia, tulee isäntäkoneet yhdistää keskenään asiaan soveltuvalla ohjelmistolla. VMwaren tapauksessa kyseinen sovellus on nimeltään vCenter Server. vCenter voidaan asentaa joko järjestelmän ulkopuolelle erilliselle fyysiselle tietokoneelle tai hallittavan järjestelmän sisään virtuaalikoneeksi. vCenter-ohjelmiston vähimmäissuorituskykyvaatimukset ovat:

- Kaksi 64-bittistä prosessoria tai yksi 64-bittinen kaksisytiminen prosessori
- Prosessorin tulee olla vähintään 2.0 GHz kellotaajuudella toimiva, ja joko Intelin tai AMD:n valmistama.

Keskusmuistia koneella tulee olla vähintään 4 GB. Muistin määrään vaikuttaa suuresti järjestelmän koko ja vCenter Serverin asetusvaihtoehdot. Mikäli vCenterin tietokanta on asennettu samalle tietokoneelle kuin itse vCenter, on muistin tarve suurempi kuin mikäli tietokanta olisi eri tietokoneella. (VMware 2014d.)

vCenter Server -ohjelmisto asennetaan erillisen käyttöjärjestelmän päälle. Käyttöjärjestelmä tulee olla 64-bittinen, ja siinä täytyy olla asennettuna Microsoft .NET 3.5 SP1 Framework. Se

asennetaan vCenter Serverin asennuksen yhteydessä, mikäli sitä ei löydy koneelta valmiiksi. Asennus vaatii internetyhteyden. Mikäli käyttöjärjestelmän kieli on muu kuin englanti, tulee samalla asentaa Microsoft .NET Framework 3.5 Language Pack. vCenter Serverin alustajärjestelmäksi sopivia käyttöjärjestelmiä ovat Microsoftin Windows Server 2008 SP2, Windows Server 2008 R2, Windows Server 2008 R2 SP1 ja Windows Server 2012. (VMware 2014f.)

Tallennustilarave on keskusmuistin tavoin riippuvainen vCenter Serverin asetusmäärittämisistä. Pienimmillään tilaa tarvitaan vain 4 GB. Mikäli kaikki vCenter Serverin ohjelmakomponentit asennetaan samalle tietokoneelle, on suositus levytilalle 100 GB. Verkko-yhteyden nopeudeksi suositellaan 1 Gb/s nopeutta. (VMware 2014d.)

Käytettäessä virtuaalista VMware vCenter Server Appliance -virtuaalisovellusta on virtuaalikoneelle annettava vähintään 70 GB levytilaa. Levytilamäärä voidaan määrittää kasvamaan automaattisesti tarpeen mukaan. Maksimikoko on 125 GB. Keskusmuistitarpeeseen vaikuttaa sovelluksella hallittavan ympäristön laajuus. Tämän työn kohdejärjestelmän ollessa VMwaren skaalan mukaan todella pieni riittää sovellukselle 8 GB minimimuisti. (VMware 2014d.)

vCenterin vaihtoehdoksi tutkittiin myös mahdollisuutta käyttää klusterin hallintaan jotakin OpenSource-ratkaisua. Mitään vartenotettavaa vaihtoehtoa ei kuitenkaan löytynyt ja koska toimeksiantajalta löytyi jo lisenssi vCenterille, tehtiin hallintatyökalun valinta perinteisen vCenter Serverin ja vCenter Server Appliancen välillä.

Tässä työssä valittiin käytettäväksi VMware vCenter Server Appliance klusterin muodostamiseen. Appliancen helppo käyttöönotto ja hallinnan yksinkertaisuus puhuivat tämän ratkaisun puolesta. Appliancen kokeiluversion sai ladattua VMwaren internetsivuilta ja asennus onnistui helposti suoraan vSphere Clientillä ESXi-palvelimelle. Asennuksessa tarvitsi tehdä vain muutamia asetusmuutoksia ja määrittää verkkoasetukset. Asennus on kuvattu käytännön osuudessa luvussa 5.1.

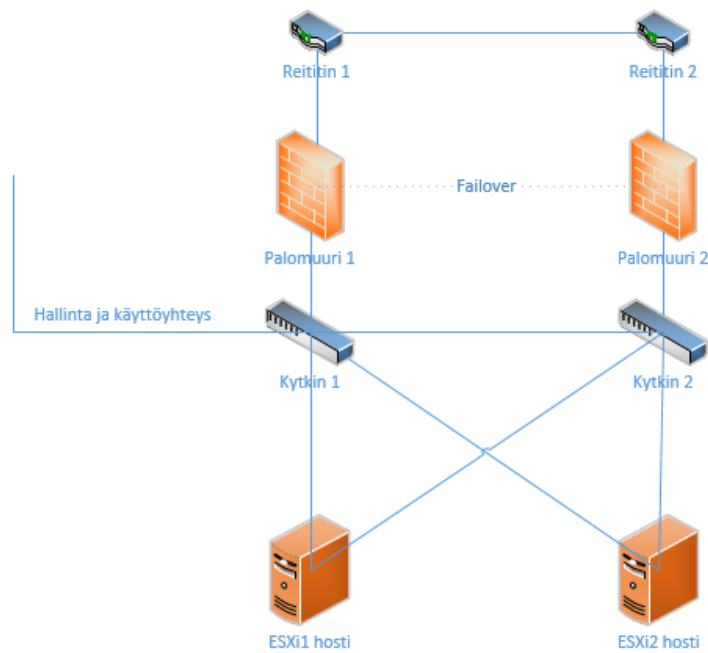
## Verkot

Kohdejärjestelmän sisäisiä verkkoja suunniteltaessa oli otettava huomioon tietoturvallisuus. VMware antaa dokumentaatioissaan hyvin tarkan kuvauksen ja ohjeistuksen, kuinka nämä verkot tulisi toteuttaa. Lähtökohtaisesti järjestelmään suositellaan määritettäväksi kolme erillistä verkkoa eri tarkoituksiin. Verkoilla on erilaiset tietoturvaominaisuudet, ja niihin pääsyä on rajoitettu käyttäjä- ja verkkotasolla. Järjestelmän sisäisiin toimiin, kuten kuormantasaukseen ja

vikasietoisuuteen, tarvitaan verkko, joka toimii vain järjestelmän sisäisenä verkkona ja on erotettu virtuaalikoneiden normaalista verkkoliikenteestä. Tätä verkkoa kutsutaan vSphere infrastruktuuriverkoksi. Järjestelmän hallintaan käytettävään hallintaverkkoon tulee rajoittaa pääsy vain ylläpitäjille. Tämä verkko suositellaan rajoitetuksi niin, että käyttöä varten tarvitaan esimerkiksi VPN-yhteys. Kolmantena verkkona on ”vm network” eli virtuaalikoneiden normaali käyttöverkko. Tämän verkon kautta tapahtuvat kaikki koneiden kommunikoinnit ulkomaailmaan ja muihin tietokoneisiin. Tässä työssä seurattiin VMwaren suosituksia ja suunnitelma oli tehdä kolme sisäistä verkkoa: ylläpitoverkko, infrastruktuuriverkko kuormantasaukseen ja tallennusjärjestelmän liikenteeseen sekä ”vm network” virtuaalikoneiden käyttöverkoksi. (VMware 2014c.)

Fyysisten palvelinlaitteiden hallinta tapahtuu Dellin etähallintajärjestelmä iDRAC:n kautta. iDRAC:ia varten oli vedetty kummastakin palvelimesta oma 1 Gb/s kuparikaapelointi ylläpitoverkkoon. ESXi-palvelinten hallintaa varten molemmista palvelimista oli vedetty 10 Gb/s nopeudella toimiva valokuitukaapelointi kahteen eri ylläpitoverkon kytkimeen. Virtuaalikoneiden käyttö tapahtuu samaa fyysistä tietoverkkoa käyttäen, sillä käyttäjämäärät ovat niin pienet, että 10 Gb/s verkko riittää hyvin ylläpito- ja käyttöyhteydeksi. Ylläpito ja käyttöverkot on erotettu virtuaaliympäristön puolella eri virtuaalikytkimen porttiryhmiin ja kytkinten puolella eri VLAN-verkkoihin. Koska palvelinten välinen vMotion-liikenne, virtuaalikoneiden siirtoyhteys isäntäkoneiden välillä, tarvitsee nopean ja ruuhkattoman kapasiteetin, tulee palvelinten välillä olla nopea kuituyhteys. vMotion sekä palvelinten ja levyjärjestelmän välinen liikenne ajetaan kummastakin isäntäkoneesta niin ikään kahdennetun 10 Gb/s valokuituyhteyden kautta. (Karppinen 2014b.)

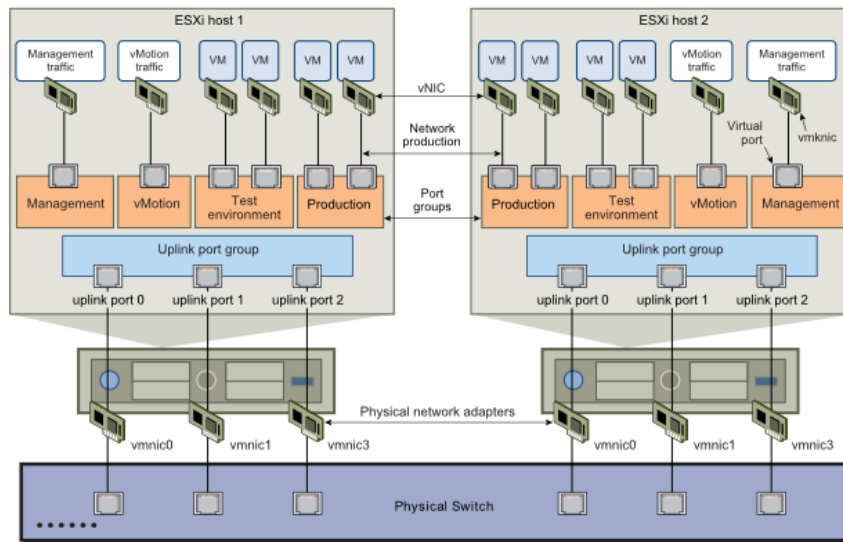
Kummastakin palvelimesta on kuvan 6 mukaisesti vedetty yhteensä kaksi valokuitukaapelia kytkimen ja palomuurin läpi kumpaankin isoon runkoreitittimeen. Näin ollen toisen reitittimen tai isäntäkoneen toisen kuitulinkin katketessa yhteys toimii edelleen toisen kuidun ja toisen runkoreitittimen kautta. Koska hallinta- ja käyttöverkko sekä kuormantasaus- ja tallennusverkko kulkevat samojen kytkinten kautta, tulee eri verkot erottaa kytkimillä omiksi virtuaali-verkoikseen (VLAN). Tällä toimenpiteellä saadaan erotettua liikenteet näennäisesti omiin verkkoihinsa, jolloin tietoliikenteen tietoturva ja hallittavuus paranee. Erillistä varmuuskopiointiverkkoa järjestelmään ei tule, koska järjestelmän käyttötarkoituksen vuoksi varmuuskopiointi ei ole tarpeellista. (Karppinen 2014b.)



Kuva 6. Hallintaverkko ja virtuaalikoneiden käyttöverkko

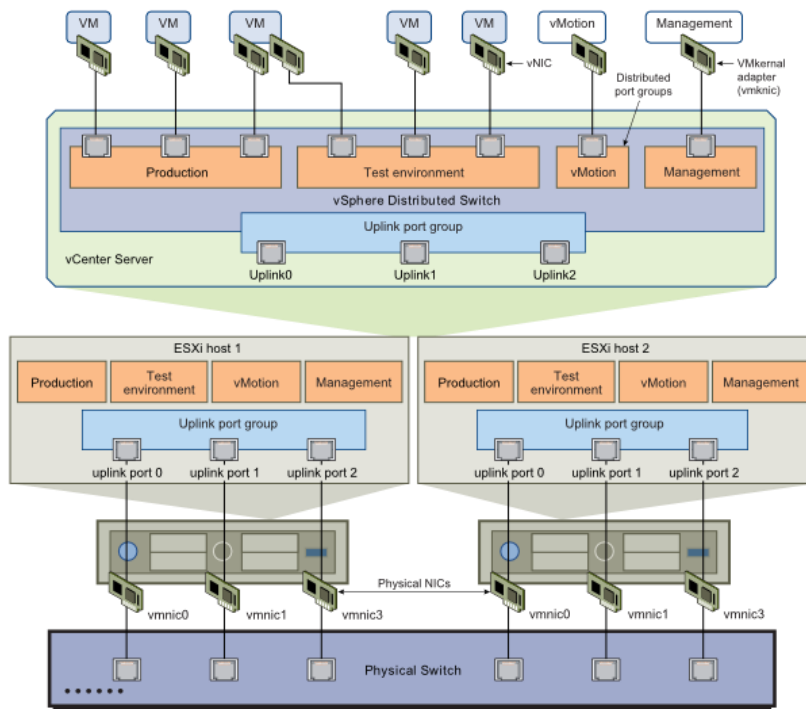
vSphere Standard Switch on VMware ESXi:n sisäänrakennettu virtuaalikytkin. Kytkimellä voidaan tehdä verkoille sama mitä virtuaalialusta tekee fyysiselle laitteistolle. Virtuaalikone näkee yhteytensä virtuaalisen kytkimen niin kuin se olisi kytketty fyysisesti verkon välityksellä kytkimeen. Virtuaalikytkimillä voidaan muodostaa verkkoja virtuaalikoneiden välille ja linkittää ne laitteiston fyysisten verkkoliitännöiden kautta ulkoiseen verkkoon. Eri verkot jaetaan virtuaalikytkimessä eri porttiryhmiin, jolloin ne ovat näennäisesti itsenäisiä verkkoja. Virtuaalikoneen verkkokortti kytketään johonkin virtuaalikytkimen porttiryhmään. Jotta virtuaalikytkimen kautta pääsisi ulkoiseen verkkoon, tulee kytkimen Uplink-portti kytkeä laitteiston fyysiseen verkkoliitännänsä. Virtuaalikoneen ja fyysisen kytkimen väliset verkotukset näkyvät kuvassa 7. (VMware 2014h.)





Kuva 7. Virtuaalikoneen ja fyysisen kytkimen väliset yhteydet Standard Switchiä käyttäen (VMware 2014h)

vSphere Distributed Switch on vCenterin myötä klusteriin tuleva ominaisuus, jolla voidaan helpottaa verkkomäärittysten tekoa klusteritasolla. Yksittäisessä ESXi-palvelimessa voidaan luoda vSphere Standard Switch -virtuaalikytkimiä paikallisen palvelimen käyttöön. Klusterin muodostamisen jälkeen on standard switchin sijaan käytännöllisempää luoda useammalle palvelimelle jaettava Distributed Switch. Distributed Switch määritellään vCenterin datacenter-tasolla. Verkon asetusmääritelmät ja hallinta tapahtuu tämän yhden virtuaalikytkimen kautta kaikille niille palvelimille, joille kytkin on jaettu. Eri virtuaaliverkot liitetään virtuaalikytkimeen eri porttiryhminä. Virtuaalikoneen ja fyysisen kytkimen väliset verkotukset näkyvät kuvassa 8. (VMware 2014g.)



Kuva 8. Virtuaalikoneen ja fyysisen kytkimen väliset verkot Distributed Switchiä käyttäen (VMware 2014g)

## Järjestelmän käyttöliittymä

Järjestelmän käyttämistä varten ei tarvita välttämättä erillisiä sovelluksia. Yleisimmät käyttötarpeet vaativat virtuaalikoneen luomisen ja siihen käyttöjärjestelmän ja testaustapausta varten tarvittavien sovellusten asentamisen. Yhteyden luomiseksi virtuaaliympäristöön tarvitaan käyttäjän koneelle VMwaren vSphere Client -sovellus. Tämäkään ei ole kuitenkaan välttämätön, sillä uusimpien VMwaren versioiden myötä ollaan siirtymässä pois perinteisestä asennettavasta asiakassovelluksesta ja vanhan vSphere Clientin tilalle tulee nykyaikaisempi internetselaimella toimiva hallinta. Virtuaaliympäristön hallintaa voi siis tehdä millä tahansa nykyaikaisella internetselaimella, johon on asennettu Adobe Flash -laajennus.

## Järjestelmän sovellukset

Ohjelmistoasennuksen osalta ensimmäinen toimenpide on asentaa fyysisille servereille VMwaren ESXi 5.5 -palvelinohjelmisto. Ohjelmiston asentamiseen on olemassa useita vaihtoehtoisia

tapoja. Eri tavat sopivat erilaisiin käyttötarkoituksiin ja käyttöympäristöihin. Eri tapoja on interaktiivinen asennus, joka tarkoittaa perinteistä CD- tai DVD-levyltä, USB-muistilta tai verkon yli PXE-käynnistyksellä tapahtuvaa asennusta. Interaktiivisessa asennuksessa asentaja joutuu tekemään asennuksen aikaiset valinnat paikan päällä. Skriptatussa asennuksessa ESXi voidaan automatisoida asennettavaksi identtisillä asetuksilla useammalle koneelle. Olemassa olevan vSphere-järjestelmän ESXi-palvelimia voidaan asentaa ja päivittää käyttäen Auto Deploy -asennusta. Tässä asennuksessa asennus hoidetaan vSphere Client -asiakasohjelmiston kautta vCenter Serveriä käyttäen. Neljäs vaihtoehto on luoda räätälöity asennuslevykuva käyttäen ESXi Image Builder CLI -komentorivityökalua. Tällä työkalulla on mahdollista räätälöidä asetukset ja lisätä asennukseen mukaan kolmannen osapuolen verkkolaite- ja levyjärjestelmäajureita. Räätälöity asennusmedia voidaan jakaa asennettavalle koneelle käyttäen asennuslevyä tai edellä mainittua Auto Deploy -toiminnallisuutta. (VMware 2014e.)

ESXi-palvelinohjelmistojen versioista voidaan valita joko VMwaren alkuperäinen ohjelmisto tai laitevalmistajan räätälöimä versio, johon on asennettu valmiiksi joitakin kyseisen valmistajan kokoonpanoon kuuluvia laiteajureita, joita VMwaren omassa asennuspaketissa ei välttämättä ole. Dellin räätälöimässä paketissa on mukana Broadcomin, Qlogicin, Intelin, Emulexin ja Dell Storage Controllerin ajureita. Dellin versiossa on tehty muutoksia joihinkin VMwaren asetustiedostoihin. Nämä tiedostot ovat /etc/vmware/oem.xml ja /etc/vmware/support. Räätälöidyn paketin mukana ei tule valmiiksi asennettuna Dell OpenManage Server Administrator -ohjelmistoa. (Dell 2014b.)

Palvelimien laitteistotason valvontaa varten päätettiin asentaa Dell OpenManage Integration for VMware vCenter 2.2 -hallintaohjelmisto. Tämä ohjelmisto mahdollistaa palvelimen laitteiston valvonnan vCenterin kautta. Valvonta helpottuu, kun fyysisen ja virtuaalisen laitteiston valvonta tapahtuu saman käyttöliittymän kautta. Dell tarjoaa ohjelmiston virtual appliancea, joten sen asennus ja käyttöönotto vSphere-ympäristössä on helppoa ja nopeaa. (Dell 2014a.)

## 5 KÄYTÄNNÖN TOTEUTUS

Määrittely- ja suunnitteluvaiheiden jälkeen alkoi itse järjestelmän käytännön toteutus. Tähän kuului fyysisten laitteiden hankinnan selvittäminen ja laitteiden valmistelu käyttökuntoon sekä ESXi -palvelinohjelmistojen ja vCenter Serverin asennus. Asennusten jälkeen suoritettiin asetusten säätäminen, testaus, käyttöönotto ja dokumentointi.

### Laitteiden hankinta

Laitteiden hankinta alkoi tiedustelulla laitehankinnoista vastaavalta tiimiltä, olisiko mahdollista saada uusia laitteita Kajaanin testausympäristöä varten. Vastauksena tuli tieto, että Kajaanissa olisi valmiina kaksi palvelinta, jotka voitaisiin irrottaa nykyisestä käytöstä ja siirtää testausympäristön käyttöön. Samoin levyjärjestelmän osalta, kyseisten palvelinten aiemmin käyttämässä levyjärjestelmässä oli reilusti vapaita resursseja testausympäristölle, joten uuden levyjärjestelmän hankinta ei ollut tarpeellista. Tämän työn myötä olemassa olevien järjestelmien käyttöasetta saatiin parannettua. Laitteiden tekniset tiedot on kuvattu luvussa 4.2.3.

### 5.1 Asennus ja asetukset

Tässä luvussa kuvataan yksityiskohtaisesti, mitä toimia laitteiston fyysisen asennuksen, palvelin- ja hallintaohjelmistojen asennuksen sekä asetusten määrittelyn aikana tapahtui. Luvussa käydään läpi myös levyjärjestelmän ja virtuaaliverkkojen asetusten määrittely. Fyysisten verkko-yhteyksien osalta toimenpiteinä oli vain kuitukaapelointia, sillä verkkoasetukset saatiin verkkoa hallinnoivalta tiimiltä.

#### Fyysinen asennus

Koska tähän työhön ei hankittu uusia palvelimia ja käyttöön otettiin olemassa olevat, eri virtualisointiympäristöstä irrotetut laitteet, ei fyysistä laitekaappiin asentamista tarvinnut tehdä. Suunnitelmissa oli asentaa uudet fyysiset palvelimet erilliseen, tätä tarkoitusta varten varattuun laitetilaan, mutta nyt laitteet saivat jäädä konesaliin, missä niiden sähkösyöttö, jäähdytys ja pa-

loturvallisuus ovat valmiiksi kunnossa. Laitteilla ajettavat palvelut voitiin siirtää ajettavaksi toisilla palvelimilla. Näin olemassa olevan järjestelmän fyysisten palvelinten käyttöastetta saatiin suuremmaksi ja parannettua konesalin energiatehokkuutta.

Käytännön työ alkoi työhön varattujen palvelinten irrottamisella olemassa olevasta järjestelmästä. Koneilla olleet virtuaalikoneet siirrettiin klusterin muille isäntäkoneille käyttäen VMwaren vMotion-migraatiotyökalua. Kun palvelimet oli saatu tyhjennettyä virtuaalikoneista ja ajettua alas, niihin lisättiin keskusmuistia ja uudet kuituverkkokaapeloinnit.

### Ohjelmistoasennukset

Tässä työssä tehtiin asennus vain kahteen fyysiseen palvelinlaitteeseen, joten erillisten skriptien tai räätälöityjen asennusmedioiden tekeminen ei ollut perusteltua. Koska kyseessä oli uusi järjestelmä, ei olemassa olevan vCenterin kautta tehtävä *Auto Deployment* ollut käytettävissä. Näin ollen asennus hoidettiin perinteistä interaktiivista asennustapaa käyttäen.

ESXi-palvelinsovellusten asennusmediaksi valittiin USB-muistitikku sen helppokäyttöisyyden vuoksi. Dellin sivuilta ladattu Dellin muokkaama VMware ESXi 5.5 -asennuslevykuva siirrettiin USB-muistitikulle käyttäen ilmaista Unetbootin-ohjelmaa. Ohjelman käyttö oli hyvin yksinkertaista. Ohjelma käynnistettiin, valittiin haluttu levykuva ja kohdemediaksi valittiin tietokoneeseen kytketty USB-muistitikku. Tämän jälkeen ohjelma purki levykuvan asennusvalmiiksi ja tallensi sen muistitikulle.

Ennen asennusten aloittamista tuli koneissa tehdä muutoksia verkkoasetuksiin. Dellin palvelinten etähallintajärjestelmä, iDRAC, tuli säätää toimimaan uusien, testiympäristöä varten määriteltyjen verkkoasetusten kautta. Vanha IP-osoite ja isäntänimi korvattiin uusilla, testiympäristölle varatuilla tiedoilla. Lisäksi koneen käynnistysmedioiden järjestykseen täytyi muuttaa USB-muistitikku ensimmäiseksi. Tämä asetus tuli tietoturvasyistä palauttaa aiemmalle tasolle ennen testiympäristön käyttöönottoa. Asennusvaiheessa annettiin asetuksen jäädä päälle, sillä tarkoituksena oli testata asennusta eri ESXi-versioilla.

### ESXi 5.5.

ESXi-palvelimen asennus onnistui Dellin räätälöimällä asennusmedialla ilman ongelmia. Ensimmäisenä valittiin kohde minne ESXi asennetaan. Tässä tapauksessa valittiin kohteeksi *Local*

*Dell PERC H310*. Näppäimistöasetteluksi valittiin *Suomi*. Viimeisenä asennuksen aikaisen asetuksena oli ylläpitäjän, root-käyttäjän, salasanan valinta. Tämän jälkeen asennusmedia piti irtottaa koneesta ja palvelin käynnistää uudelleen, jolloin kone käynnistyi uuteen ESXi-ympäristöön.

Asennuksen jälkeen täytyi määritellä ESXi:n hallintaverkon asetukset, jotta palvelinta pystyisi käyttämään verkon yli vSphere Clientilla. Asetuksia pääsi muokkaamaan painamalla F2-näppäintä palvelimen käynnistyttyä ja syöttämällä root-käyttäjän salasana. Hallintaverkon asetukset määriteltiin kohdasta **Configure Management Network**. Valittiin verkkoliitännöiksi fyysiset verkkokortit *vmnic5* ja *vmnic7*, jotka ovat palvelimeen lisättyjä valokuituliitäntöjä. Näistä liitännöistä lähtevät hallintayhteydet valokuitukaapelina kahdelle eri hallintaverkon reitittimelle. Fyysisten verkkokorttien osalta on mietitty myös vikasietoisuutta. Sen vuoksi hallintaverkko ei käytä samalla fyysisellä verkkokortilla olevia *vmnic5*- ja *vmnic6*-liitäntöjä, vaan liitännät ovat eri korteilla. Näiden kanssa ristiin menee Storage ja vMotion -liikenne, joka kulkee toista kuituparia myöten porteissa *vmnic4* ja *vmnic6*. **IP configuration** -valikosta asetettiin palvelimelle varattu staattinen IP-osoite. **IPv6 Configuration** -kohdasta poistettiin käytöstä IPv6-protokolla, koska sitä ei hallintaverkon puolella tällä hetkellä tarvita. **DNS configuration** -valikossa määriteltiin hallintaverkon DNS ja Hostname -asetukset. Asetusten säädön jälkeen piti palvelin käynnistää uudelleen, koska IPv6 muutokset vaativat uudelleenkäynnistyksen. Tämän jälkeen palvelin oli valmis käytettäväksi hallintaverkon kautta vSphere Client-hallintasovelluksella.

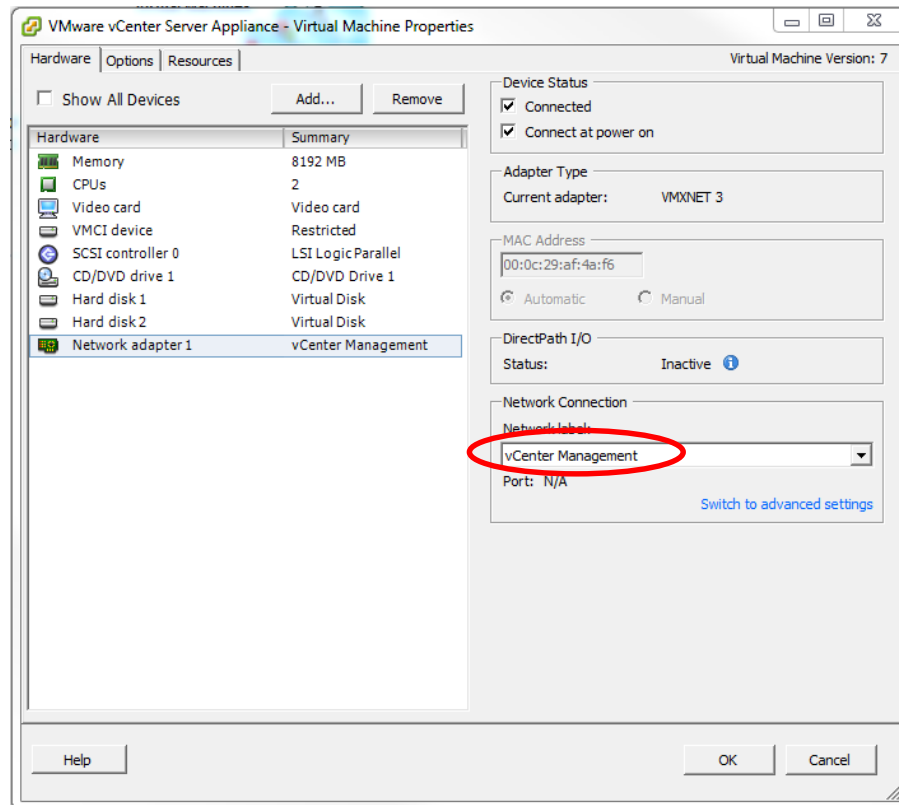
Toiselle palvelimelle asennettiin vertailumielessä VMwaren oma versio ESXi 5.5.:stä. Asennuksessa vertailtiin asennusprosessia ja mahdollisia muita näkyviä muutoksia. Asennusprosessi meni aivan kuin Dellin räätälöimällä levykuvalla, eikä vSphere Clientilla palvelimia vertaillen näkynyt muita eroavaisuuksia kuin palvelinohjelmiston versionimi ja -numero. Eroavaisuudet huomattaisiin käytännössä vain, mikäli palvelimeen olisi asennettuna sellaisia verkko- tai tallennuslaitteistoja, joita VMwaren alkuperäinen ESXi-asennusmedia ei tukisi.

### vCenter Server Appliance

vCenter Server asennettiin virtual applanacena eli virtuaalisena valmissovelluksena. Tämä on helpompi ratkaisu perinteiseen menetelmään verrattuna, jossa ensin asennetaan alustapalvelin eli jokin Microsoft Windows Server -versio, ja sen päälle vCenter Server -ohjelmisto. Virtual

Appliancen asennus oli hyvin yksinkertaista. vSphere Clientillä otettiin yhteys ESXi-palvelimeen, jolle sovellus haluttiin alun perin asentaa. vSphere Clientissä avattiin **File**-valikon *Deploy OVF Template* -työkalu. Avautuneeseen ikkunaan haettiin VMwaren internetsivuilta ladattu tiedosto ja painettiin **Open**. Avautuneeseen asetustenmäärittelyikkunaan syötettiin vCenter Server Appliencelle haluttu nimi ja valittiin **Next**. Oletuksena tarjottu *VMware vCenter Server Appliance* -nimi oli hyvä, eikä sitä nähty tarpeelliseksi muuttaa. Seuraavalla sivulla valittiin haluttu tallennussijainti eli *datastore*. Koska tällä hetkellä palvelinta ei ollut liitetty mihinkään ulkoiseen levyjärjestelmään, tuli sovellus asentaa palvelimen sisäiselle levyille. Appliencelle varattavan levytilan tyyppiä valittiin *Thick Provision Lazy Zeroed*, joka oli oletusvalinta. Tämä tarkoittaa lähinnä sitä, että levyltä varataan suoraan koko 125 GB, minkä verran appliancen määritykset haluavat, eikä levytilaa pyritä säästämään varaamalla levyltä tilaa vain tarvittava määrä ja kasvattamalla sitä tarpeen mukaan maksimimäärään saakka. Lopuksi tarkastettiin annetut määritykset yhteenvetönäkymästä ja painettiin **Finish**. Tässä vaiheessa jätettiin valitsematta *Power on after deployment* -kohta, sillä luodulle virtuaalikoneelle piti tämän ympäristön tapauksessa määrittellä manuaaliset verkkosovitinasetukset, eikä tämä ollut mahdollista asennustyökalun avulla.

vCenter Appliancen virtuaalikoneen verkkoasetuksia piti muuttaa ennen kuin vCenter oli mahdollista asentaa loppuun. Koska virtuaaliympäristön hallintaverkko oli tehty VMkernel-adapterina, sitä ei voinut näyttää suoraan vCenter-koneen verkkokortiksi. Hallintaverkolle piti lisätä *Virtual Machine Port Group* -porttiryhmä samalla VLAN ID:llä kuin hallintaverkko. Tämän jälkeen kyseinen porttiryhmä voitiin asettaa vCenterille verkkosovittimeksi (kuva 9). vCenter yrittää oletuksena saada IP-osoitteen DHCP-palvelimelta, mutta koska sellaista ei tässä verkossa ole, tuli palvelimen IP-osoitteeksi 0.0.0.0.



Kuva 9. vCenterille lisätty uusi vCenter Management -verkko

Ensimmäisenä toimenpiteenä vCenter Applianceen käynnistyttyä tuli määritellä sille kiinteä IP-osoite vSphere Clientin konsolinäkymän kautta. Tämä toimenpide oli pakko tehdä ensimmäisenä, sillä vCenterin asennus viimeistellään internetselaimella. IP-määritysten asettaminen tapahtui vami\_config\_net -työkalun avulla (Beerens, I. 2013). Kirjautuminen järjestelmään tapahtui oletuskäyttäjätunnuksella *root* ja salasanalla *vmware*. Järjestelmälle annettiin komento: */opt/vmware/share/vami/vami\_config\_net* (kuva 10). Vami-työkalulla säädettiin *Gateway*, *Hostname*, *DNS* ja *IP osoite* oletusverkkosovittimelle *eth0*.

```
localhost:~ # /opt/vmware/share/vami/vami_config_net
Main Menu
0) Show Current Configuration (scroll with Shift-PgUp/PgDown)
1) Exit this program
2) Default Gateway
3) Hostname
4) DNS
5) Proxy Server
6) IP Address Allocation for eth0
Enter a menu number [0]:
```

Kuva 10. Vami\_config\_net -ohjelman valikko (Beerens, I. 2013)



IP-osoitteen asettamisen jälkeen vCenter Serverin käyttö aloitettiin asettamalla tietokanta ja kirjautumisasetukset internetselaimen avulla. Selaimella tulee ottaa HTTPS-yhteys edellä määritelyyn vCenterin IP-osoitteeseen portin 5480 kautta. Kun asetukset oli säädetty ja vCenter käynnistetty, pääsi internetselaimella kirjautumaan vCenter Web Clientiin. Tämä internetse-  
laimella toimiva hallintatyökalu on vCenter 5.5. -versiosta asti ollut suositeltu ja pääasiallinen käyttöliittymä vSphere-ympäristön hallintaan. (VMware 2014a.)

#### ESXi-palvelinten asetukset

Suurin osa järjestelmän asetusmäärittelyistä tehtiin vasta vCenter Serverin asennuksen jälkeen keskitetysti. Ennen klusteriksi liittämistä palvelimille tehtiin kuitenkin joitakin asetusmäärittelyitä, jotta palvelinten yhdistäminen klusteriksi mahdollistuisi.

Palvelimissa käynnistettiin SSH-palvelu ja määriteltiin se käynnistymään aina palvelimen mukana. Tätä tarvittiin muun muassa verkkoyhteyksien testaamiseen ennen vCenter Serverin asennusta ja klusterin luomista. SSH-palvelun käynnistys tapahtui vSphere Clientillä *Configuration*-välilehdeltä *Security Profiles* -näkyvän *Services Properties* -asetusikkunasta.

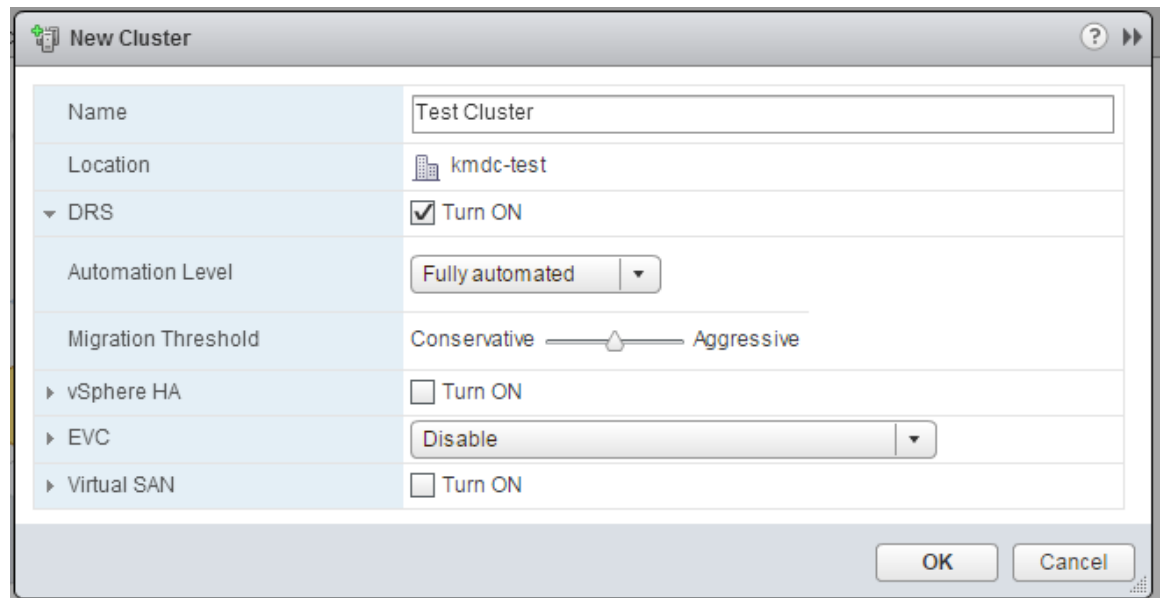
Palvelinten pysymiseksi samassa ajassa määriteltiin molempien ESXi-palvelinten asetuksiin aikapalvelin. Aikapalvelimeksi määritettiin CSC:n Funet-verkossa oleva ntp1.funet.fi -palvelin. Määrittäminen tapahtuu vSphere Clientillä ESXi-palvelimen *Configuration*-valikosta *Time Configuration* -kohdasta. Ajan synkronoitumiseen meni hetki aikaa.

Jotta vCenter Serverin asennus onnistuisi, tuli palvelimen ylläpitoverkkoon määritellä oma porttiryhmä vCenterin hallintaan. Oletuksena luotu vSwitch0-virtuaalikytkin on oletuksena määritelty sekä virtuaalikoneiden käyttöyhteydeksi että hallintaverkon käyttöön. vCenterin käyttö olisi onnistunut virtuaalikoneiden käyttöyhteyden kautta, mutta hallinta haluttiin eriyttää käyttöverkosta, joten vCenterin hallintaan tuli luoda uusi porttiryhmä. Virtuaalikytkimiin määriteltiin myös fyysisen verkon puolella käytettävät virtuaaliset lähiverkot, VLAN:it.

#### Yksittäisistä palvelimista klusteriksi

Palvelinklusterin luominen aloitettiin vCenter Server Web Clientin kautta. Vasemman laidan valikosta valittiin **vCenter** -> **Hosts and Clusters** -> **Create Datacenter**. Nimeksi asetettiin *kmdc-test*. Datacenterin alle luotiin kuvan 11 mukaisilla asetuksilla klusteri nimeltään *Test Cluster*.

Klusterin asetuksissa otettiin aluksi käyttöön DRS-ominaisuus oletusasetuksillaan. DRS tarkoittaa dynaamista resurssien tasausta. Järjestelmän korkeaa käytettävyyttä takaava, vikatilanteista toipumisen työkalu, vSphere HA otetaan käyttöön järjestelmän ollessa käytössä, mikäli sitä jossakin käyttötapauksessa tarvitaan.



Kuva 11. Klusterin asetukset

Klusterin luonnin jälkeen voitiin alkaa lisäämään ESXi-palvelimia klusterin jäseniksi. Palvelimen lisääminen oli hyvin suoraviivaista. Ensin annettiin halutun palvelimen DNS-nimi, syötettiin ylläpitokäyttäjätunnukset ja valittiin lisenssiksi kokeilulisenssi, kuten aiemmin ESXi-palvelimia asennettaessakin tehtiin. Palvelimen *Lockdown modea*, joka estää palvelinten suoran hallinnan vSphere Clientillä vCenteriin liittämisen jälkeen, ei otettu käyttöön, sillä mikäli jotain menee esimerkiksi verkkoasetuksissa pieleen, eikä vCenterin kautta pystykään enää hallitsemaan palvelinta, estäisi *Lockdown Mode* yhteydenottamisen suoraan ESXi:n omaan hallintaan. vSphere-ympäristössä on mahdollista jakaa fyysisten palvelinten laskentaresursseja useampaan *resurssipooliin*. Resurssipoolia ei ollut luotu ESXi-palvelimille, joten tässä yhteydessä otettiin käyttöön klusterin oma resurssipooli, jolloin käytössä on kaikki järjestelmän kapasiteetti.

#### Levyjärjestelmän määrittely ja liittäminen klusteriin

Levyalueiden luonti Dell EqualLogic PS6100E -levyjärjestelmään onnistui yksinkertaisesti. Hallintatyökalussa valittiin vasemman laidan valikosta **Volumes**, jonka pikatoiminnoista löytyi **Create Volume**. Auenneessa ikkunassa annettiin halutun levyalueen nimi, kuvaus ja valittiin,

mistä levyjärjestelmän resurssipoolista levyä varataan. Sen jälkeen säädettiin haluttu koko. Testausympäristöä varten luotiin aiemmin määritellyt 2 TB nopeaa RAID 50 -levyä, 2 TB hitaampaa RAID 6 -levyä ja hidas 4 TB RAID 6 -levy. Levyjärjestelmän päässä ei otettu käyttöön tilaa säästävää *thin provisioning* -asetusta, sillä tämä hoidetaan virtuaaliympäristön päässä. Kaikkiin alueisiin jätettiin oletuksena tarjottu *100 % snapshot reservation*. Pääsy rajattiin IP-osoitteiden ja CHAP-autentikoinnin avulla. Jotta samaa levyaluetta voidaan käyttää yhtä aikaa molemmilta palvelimilta, tuli valita asetus *Allow simultaneous connection from initiators with different IQNs*. Sen jälkeen määriteltiin vielä levyalueen oikeuksiksi sekä luku- että kirjoitusoikeudet. Tämän jälkeen levyalueen luonti oli valmis. Tosin EqualLogic jakoi levyalueille levyä kummastakin erinopeuksisesta resurssipoolista, joten asetusta piti muuttaa käsin ennen levyjen käyttöönottoa. Tämä tapahtui levyalueen *Modify settings* -näköymästä. Täältä valittiin *Raid preferences* -kohtaan RAID 6 tai RAID 50 levyalueen käyttötarkoituksen mukaan.

EqualLogic-levyjärjestelmän hallintapaneelissa määriteltiin CHAP-autentikointiprotokolla käyttöön juuri luoduille levyalueille. Ensin luotiin CHAP-käyttäjä hallintapaneelin **Group Configuration -> iSCSI Authentication -> Local CHAP Accounts** -näköymässä. Uudelle käyttäjälle luotiin nimi ja salasana, eli CHAP secret. Varmistettiin, että luotavan käyttäjän tila on *enabled*. Tämän jälkeen voitiin aktivoida CHAP-autentikointi halutuille levyalueille lisäämällä juuri luotu CHAP-käyttäjänimi levyalueen Access-määrittäjiin.

Kun levyalueet ja niiden pääsymäärittelyt oli luotu, voitiin levyalueet ottaa käyttöön klusterissa. Ensin kokeiltiin levyn liittämistä pelkästään toiselle palvelimelle. Tämä tapahtui esx1-palvelimen *Manage*-välilehdeltä määrittämällä tallennustila-asetuksiin uusi ohjelmallinen iSCSI-sovitin (iSCSI Software Adapter). Sovittimen ilmestyttyä listaan luotiin sille autentikointimäärittelyt. Määrittelyt löytyvät sovitin *Properties*-välilehdeltä *Authentication*-kohdan *Edit*-napin alta. Autentikointitavaksi valittiin *Unidirectional CHAP if required by target*, kuten kuvassa 12 näkyy. Nimeksi ja salasanaksi syötettiin aiemmin levyjärjestelmään luodut CHAP-autentikointitiedot. Tämän jälkeen voitiin luoda yhteys levyjärjestelmään sovitin *Targets*-välilehdeltä.

vmhba38 - Edit Authentication

The initiator uses these settings for authentication for all targets unless otherwise overridden by the specific target settings. Make sure that these parameters match on the storage side.

Authentication Method: Use unidirectional CHAP if required by target

**Outgoing CHAP Credentials (target authenticates the initiator)**

Name: ☐ Use initiator name  
testesx

Secret: \*\*\*\*\*

**Incoming CHAP Credentials (initiator authenticates the target)**

Name: ☐ Use initiator name

Secret:

OK Cancel

Kuva 12. CHAP-määrittysten luonti iSCSI-initiaattorille.

vCenterin kiinteä levyjärjestelmän etsiminen, *static discovery*, ei löytänyt ensin levyjärjestelmän IP-osoitteesta mitään laitteita, mutta dynaaminen etsintä, *dynamic discovery*, löysi. Dynaamisen etsinnän jälkeen löytyi myös *static discovery* -puolelle kaikki levyjärjestelmän LUN:it. Dynaaminen levyjärjestelmän skannaaminen osoittautui huonoksi ratkaisuksi, sillä palvelin skannasi kaikki levyjärjestelmän osiot, ja koska iSCSI-sovittimelle oli määritelty testausympäristön CHAP-autentikointitunnukset, tulkitsi levyjärjestelmä palvelimen skannauksen tunkeutumisyriyukseksi ja hälytti luvattomasta yhteysyriyksestä. Tämän jälkeen jatkettiin levyjen liittämistä *static discoveryä* käyttäen. Staattisesti levyjä liitettäessä tulee syöttää levyjärjestelmän IP-osoite ja LUN:in IQN-osoite. CHAP-määrittelyt määriteltiin perittäväksi iSCSI-initiaattorilta.

Kun yhteys levyjärjestelmään oli luotu, tuli iSCSI-sovittimelle luoda tallennusverkko. Tämä tapahtui sovittimen *Networking*-välilehdeltä. VMkernel-sovittimien näkymään luotiin uusi VMkernel-sovitin, joka määriteltiin uudelle, samalla luotavalle vSwitch1:lle. Verkon nimeksi valittiin iSCSI network, VLAN ID:ksi määritettiin 100 ja Enabled Services -kohtaan valittiin Virtual SAN traffic. Seuraavaksi syötettiin verkkotiimiltä saadut iSCSI-verkon IP-osoitetiedot

ja sovittimen luonti oli valmis. Seuraavaksi voitiin lisätä juuri luotu iSCSI network aiemmin luodun iSCSI-sovittimen verkkoyhteydeksi *Network Port Binding* -välilehdeltä. Kun verkko oli lisätty, ajettiin palvelimen tallennusverkon adapttereille uusi skannaus, jolloin järjestelmä tarkistaa asetukset ja etsii uusia verkkolevyjä. Skannauksen jälkeen ilmestyivät levyjärjestelmään testiympäristölle luodut levyalueet *Devices*-välilehdelle.

Kun iSCSI-yhteys levyalueisiin saatiin muodostettua, tuli levyalueet liittää palvelinten tietovarastoiksi (*datastore*). Tämä tapahtui vSphere Web Clientin kautta palvelinten *Related Objects* -välilehtien *Datastores*-näkymistä. Levyalue saadaan lisättyä datastoreksi painamalla *Create new datastore*. Kun datastore lisätään, on oletussijainti kyseinen isäntäkone. Ensimmäiseksi levyalueeksi tehtiin ”Images datastore”. Tähän on nimensä mukaisesti tarkoitus tallentaa käyttöjärjestelmien levykuvia. Levyalueen tyyppi: VMFS, Nimi: ”Images datastore”. Palvelimelle näkyvät LUN:it listautuivat alapuolelle, josta valittiin oikea tätä datastorea varten luotu LUN. VMFS -versioksi otettiin VMFS 5, koska se tukee myös yli 4TB LUN:eja. Osioinniksi valittiin yksi koko LUN:in kokoinen osio.

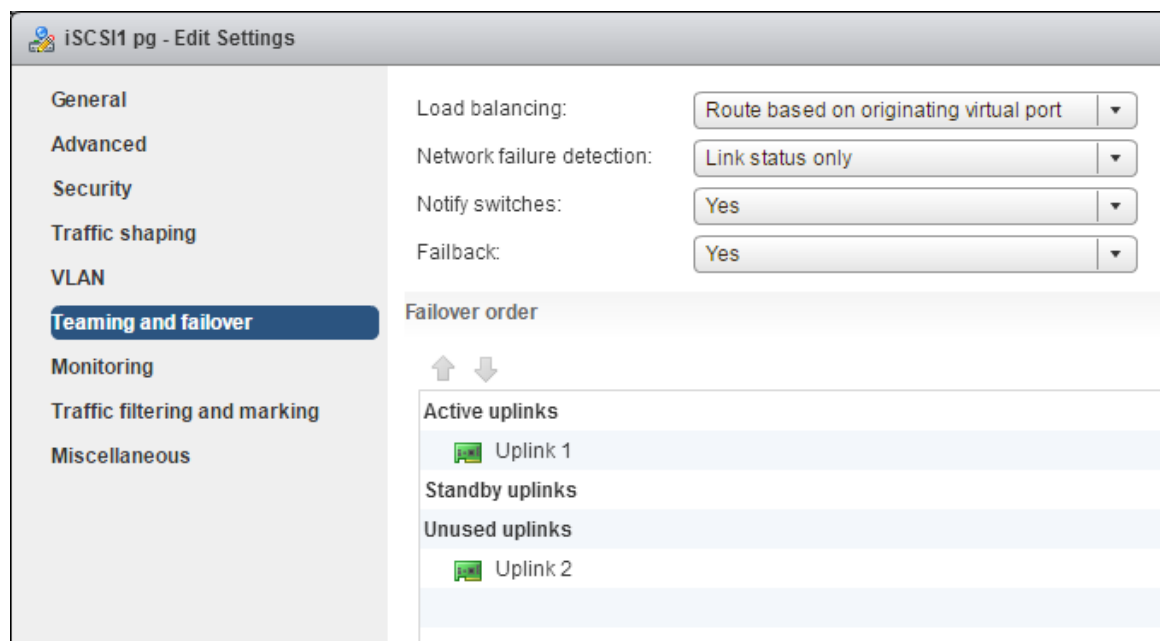
Samalla muutettiin palvelimen paikallinen kiintolevy näkymään nimellä ”ESX1 local datastore”, jotta se erottuisi yhteisistä levyalueista. Tämä toimenpide tehtiin myöhemmin myös esx2-palvelimelle.

Ensimmäisen palvelintasolla luodun levyalueen jälkeen luotiin klusteriin Distributed Switch. Distributed switchin luonti kuvataan seuraavassa kappaleessa. Tämän myötä helpottui tallennusverkon asetusten säätäminen yhtä aikaa molemmille palvelimille. Distributed switchin ja porttiryhmiä luonnin jälkeen esx1-palvelimelle luotu ”Images datastore” ilmestyi automaattisesti myös esx2-palvelimelle. Loput kaksi levyaluetta luotiin distributed switchin mahdollistamana klusteritasolla. Muut kaksi levyaluetta ovat nimeltään ”VM datastore”, jonne tallennetaan virtuaalikoneiden tiedostot ja ”Storage datastore”. Tämä on isompi, 4 TB kokoinen levyalue, jolta voi varata virtuaalikoneille lisää tallennustilaa.

Distributed switchin luominen tapahtui valitsemalla vCenter Server Web Clientin vasemmasta laidasta luotu datacenter. Yksinkertaisimmillaan uuden distributed switchin luontiin pääsi painamalla datacenteriä hiiren oikealla napilla ja valitsemalla **New Distributed Switch...** Ensimmäisenä annettiin luotavalle kytkimelle nimi. Tässä tapauksessa luotiin kytkin infrastruktuuriverkon käyttöön. Infrastruktuuriverkon kytkimelle annettiin nimeksi ”Storage and vMotion sw”. Seuraavaksi valittiin distributed switchin versio. Tämä versio on riippuvainen klus-

terissa olevien palvelinten versioista. Koska tässä järjestelmässä ESXi-palvelimet ovat 5.5.-versioita, valittiin distributed switchin versioksi 5.5.0. Kytkein ei ole alaspäin yhteensopiva vanhempien palvelinversioiden kanssa. Kytkimen uplink-porttien määräksi valittiin kaksi, koska palvelimissa on kaksi fyysistä verkkoliitäntää infrastruktuuriverkkojen käyttöön. *Network I/O Control* annettiin olla oletusasetuksessaan käytössä. *Default port group* -kohdassa annettiin olla ruksi, sillä samalla haluttiin luoda oletusporttiryhmä. Porttiryhmän nimeksi määriteltiin ”iSCSI1 pg”.

Distributed switchin luonnin jälkeen tuli määritellä kytkimen porttiryhmät. Kytkeä luodessa luotiinkin oletusryhmä, mutta porttiryhmiä tarvitsi luoda lisää ja määritellä luodun ryhmän asetuksia. Porttiryhmien määrittely tapahtui datacenter-näkymän *Related Objects* -välilehden *Distributed Port Groups* -sivulta. Koska kyseistä kytkintä käyttävät sekä iSCSI-tallennusliikenne että vMotion-kuormantasaus, tulevat verkot erottaa toisistaan porttiryhmillä ja VLAN:illa. Kytkeä luodessa tehty porttiryhmä ”iSCSI1 pg” määriteltiin loppuun määrittämällä sen VLAN ID:ksi 100 ja vikasietoisuusasetuksissa (failover) määriteltiin Uplink1 olemaan aktiivinen ja Uplink2 passiivinen. Vikasietoisuuden toimivuuden vuoksi kytkimelle tehtiin toinen porttiryhmä nimeltään ”iSCSI2 pg”, muuten identtisillä asetuksilla, paitsi Uplink1 on passiivinen ja Uplink2 aktiivinen. iSCSI1 pg:n failover-määrittelyt näkyvät kuvassa 13. Molempiin iSCSI-porttiryhmiin määriteltiin liikenteen tyypiksi *Virtual SAN*. ”vMotion pg” -porttiryhmälle määriteltiin vain VLAN id 85 ja liikenteen tyypiksi vMotion. (VMware 2013a. 14–15.)



Kuva 13. Tallennusverkon iSCSI1 pg -porttiryhmän failover-määrittely

Kun porttiryhmät olivat luotu, oli aika jakaa kytkin palvelimille. Tämä tapahtui klusterin **Distributed Switches** -näköymästä. Näköymästä valittiin haluttu kytkin ja **Actions**-valikosta valittiin **Add and Manage Hosts...** *Add hosts* -valinnalla päästiin valitsemaan klusterissa olevista palvelimista, mille kytkin haluttiin jaettavaan. Tässä tapauksessa valittiin molemmat palvelimet. Seuraavaksi valittiin, mitä toimenpiteitä palvelinten lisäyksen yhteydessä haluttiin tehdä. Storage and vMotion sw:n osalta valittiin *Manage physical adapters*, koska kytkimelle haluttiin määrittellä palvelinten fyysiset uplink-verkkoliitännät. Uplinkeiksi valittiin kummaltakin koneelta *vmnic4* Uplink1:ksi ja *vmnic6* Uplink2:ksi. Valinnan jälkeen järjestelmä analysoi, onko tehdyillä määrityksillä vaikutusta olemassa oleviin verkkoihin. Tämän jälkeen oltiin valmiita suorittamaan muutokset.

#### Virtuaalikoneiden käyttöverkon luonti

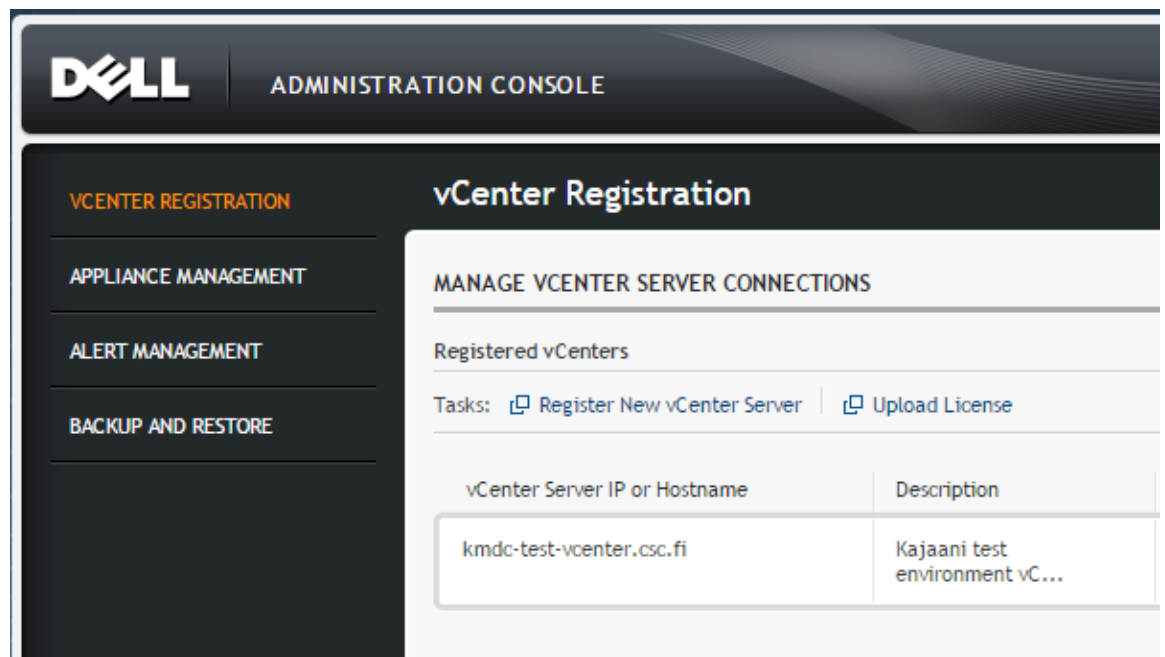
Virtuaalikoneiden käyttöverkon porttiryhmäksi määriteltiin ”VMnetwork pg”. Tämä tehtiin perinteistä vSwitch Standard Switch -virtuaalikytkintä käyttäen kummallekin palvelimelle erikseen. Palvelimen asetuksista valittiin **Manage**-välilehti, **Networking** ja **Virtual switches**. Täältä painettiin **Add host networking**. Koska tarkoituksena oli luoda verkko virtuaalikoneiden käyttöyhteydeksi, valittiin **Virtual Machine Port Group for a Standard Switch**. Koska kyseiset fyysiset verkkoliitännät olivat jo hallintaverkon käytössä *vSwitch0*:lla, lisättiin uusi porttiryhmä olemassa olevalle virtuaalikytkimelle. Porttiryhmän nimeksi määriteltiin ”VMnetwork pg” ja VLAN ID:ksi 3202. Kun porttiryhmä oli tehty molemmille palvelimille, pystyttiin kyseinen porttiryhmä määrittelemään virtuaalikoneen verkkoliitännäksi. Porttiryhmän toiminta testattiin määrittelemällä virtuaalikoneelle IP-osoite ja testaamalla, että koneella pääsee verkkoon ja Internetiin. Porttiryhmän asetusten identtisyys palvelinten välillä testattiin siirtämällä virtuaalikone palvelimelta toiselle, ja koska yhteys toimi, olivat asetukset molemmilla palvelimilla samanlaiset.

#### OpenManage Integration for VMware vCenter

Kiintolevyjen ja muun fyysisen laitteiston valvonta haluttiin yhdistää virtuaaliympäristön valvontaan. Tämä oli mahdollista Dellin laitteiston osalta käyttäen Dellin luomaa OpenManage Integration for VMware vCenter -työkalua. Työkalua saa kokeilla ilmaiseksi 90 päivän ajan enintään viiden ESXi-palvelimen ympäristössä. Työkalun asentamista varten täytyi ladata Del-

lin sivuilta pakattu tiedosto, jonka sisältä löytyi suoritettava tiedosto. Tämä tiedosto oli suoritettava ennen virtual appliancen asennusta, koska käyttöehdot olivat asennuspaketin sisällä. Ehdot hyväksyttyä paketti purki itsensä ja sisältä löytyi appliancen sisältävä tiedosto. Tämä tiedosto ladattiin toiselle palvelimelle, samoin kuin aiempi vCenter Server virtual appliance. Dellin ohjeistukset suosittelevat käytettäväksi kiintolevyn asetuksiin *Thick Provision Eager Zeroed* -määrittystä. Appliancen laiteasetuksia määriteltäessä ei pystynyt valitsemaan jostakin syystä vCenter Management -verkkoa, johon kyseinen palvelu tuli liittää, jotta sen yhteydet ESXi-palvelimiin ja vCenteriin toimisivat. Tämän vuoksi laiteasetuksia tuli muuttaa ennen palvelun käynnistämistä. (Dell 2014c.)

Käynnistyksen jälkeen tuli määrittää verkkoasetukset ja ylläpitäjän salasana, samoin kuin vCenter virtual appliancen kanssa. Konsolin kautta tehtävien määritysten jälkeen pääsi asennuksen viimeistelemään internetselaimen kautta. Selainpohjaisessa hallinnassa painettiin **Register New vCenter Server** ja syötettiin vCenterin osoite sekä kirjautumistiedot. Koska ohjelma asennettiin testiversiona, voitiin lisenssin syöttövaihe ohittaa. Kuvassa 14 näkyy työkaluun rekisteröity testiympäristön vCenter.

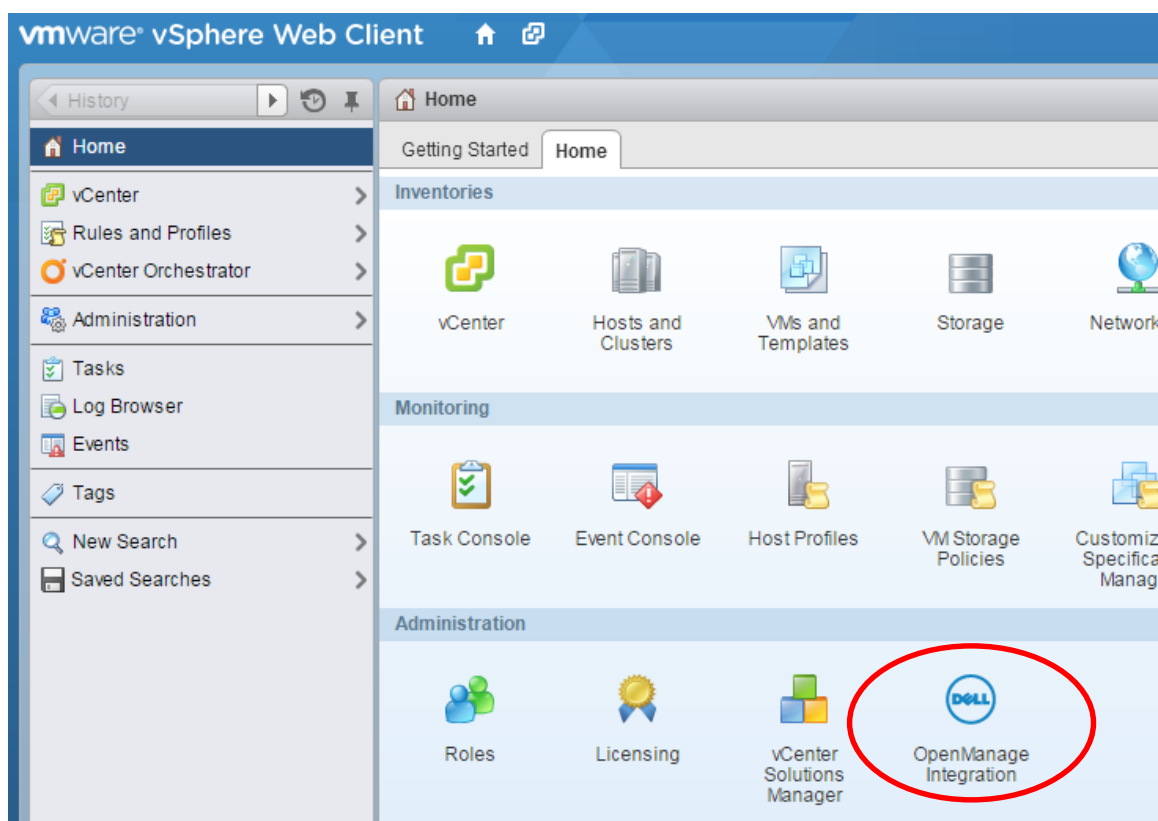


Kuva 14. OpenManage Integration -työkaluun rekisteröity testiympäristön vCenter

vCenterin rekisteröinnin jälkeen vCenter Web Clientin hallintanäkymään ilmestyi uusi kuvake ”OpenManage Integration” (kuva 15). Työkalun käynnistyessä ensimmäisen kerran avautui ohjattu asetustyökalu. Työkalussa valittiin valvottavien palvelinten määrittelyt. Ensimmäinen työkalu ilmoitti, että asetustyökalu tulee ajaa kaikilla järjestelmään kytketyillä vCenter-koneilla. Tässä



tapauksessa siis yksi kerta riitti. Seuraavaksi luotiin yhteysprofiili palvelimia varten. Profiiliin määriteltiin palvelinten iDRAC-hallintayhteyden tunnukset ja ESXi-hallintatunnukset. Jotta OpenManage Integrationin näyttämä tieto laitteistosta pitäisi paikkansa, Dell suosittelee ajamaan järjestelmän inventaarion säännöllisesti. Sopiva väliaika tarkistuksille on asetustyökalun mukaan viikko. Inventaario määriteltiin ajettavaksi kerran viikossa, perjantai-iltaisin kello 18. Työkalun avulla voisi tarkkailla myös laitteiston takuun voimassaoloa. Tätä varten voitaisiin tehdä myös ajoitettu tarkistus. Tässä ympäristössä tätä ominaisuutta ei otettu käyttöön, sillä takuuasiat hoidetaan muuta kautta. Työkalun avulla voidaan välittää laitteistotason hälytyksiä vCenteriin, jolloin vCenter voi ajaa vikatilanteessa laitteiston hallitusti alas. Tätä ominaisuutta ei suositella otettavaksi käyttöön, ellei järjestelmään ole määriteltä DRS-ominaisuutta, sillä muuten palvelimella ajettavat virtuaalikoneet sammuisivat odottamatta. Vaikka tässä ympäristössä onkin toimiva DRS-määrittely, ei ominaisuutta otettu vielä tässä vaiheessa käyttöön. Ominaisuus vaatii testaamista ennen käyttöönottoa. Virtualisointiin liittyvät ilmoitukset otettiin kuitenkin käyttöön. (Dell 2014c.)



Kuva 15. OpenManage Integrationin pikakuvake vCenterin aloitusnäkyssä

## 5.2 Testaus

Ensimmäinen testi järjestelmän toiminnalle tehtiin heti levyjärjestelmän liittämisen ja vMotion-siirtoverkon luomisen jälkeen. Aiemmin paikallisesti ensimmäiselle palvelimelle asennetun vCenter Serverin tiedostot piti siirtää jaettuun levyjärjestelmään. VMwaren tapauksessa tämä oli helppo suorittaa. vCenter Web Clientissä valittiin halutun virtuaalikoneen päällä **Migrate...** ja avautuneesta ikkunasta valittiin **Change datastore**. Tämän jälkeen valittiin haluttu jaettu levyalue. vCenter Serverin data siirtyi ongelmitta paikalliselta levyltä jaetun levyjärjestelmän virtuaalikoneille tarkoitetulle levyalueelle. Siirron aikana ei tapahtunut käyttökatkoa.

Seuraava testi, järjestelmän kuormantasauksen toimivuus, tehtiin toisen palvelimen klusteriin lisäämisen jälkeen. Kun toiseenkin palvelimeen oli määritelty hallinta- ja tallennusverkot, testattiin virtuaalikoneen ajonaikaista siirtoa palvelinten välillä. Testikoneena toimi tälläkin kertaa vCenter Server. Tämän vuoksi voitiin samalla testata, tuleeko siirron aikana käyttökatkoa, koska siirto komennettiin vCenter Serverin Web Clientistä. Palvelinten välillä tapahtuva siirto suoritetaan saman työkalun kautta kuin aiemmin tehty datastoren siirto.

### DRS-testi

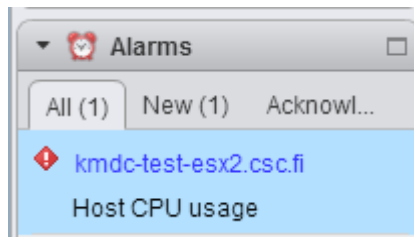
Dynaamisen resurssien tasauksen (DRS) testaamista varten luotiin klusteriin viisi Ubuntu Linux 14.04 -virtuaalikonetta. Koneisiin asennettiin cpuburn-sovellus, jolla voidaan luoda kovaa laskentakuormaa koneen prosessorille. Cpuburn-sovelluksen asennus tapahtui komentamalla terminaalissa `sudo apt-get install cpuburn`. Kuorman tarkkailuun asennettiin työkalu glances. Glances käynnistettiin komennolla `glances`. Prosessorille luotiin kuormaa komennolla `burnP6`. Kaikissa viidessä testiä varten luodussa Ubuntu Linux 14.04 -virtuaalikoneessa käynnistettiin kaksi cpuburn-prosessia ja todennettiin kuorman syntyminen glances-ohjelmalla (kuva 16). Kaikki virtuaalikoneet oli ennen testiä siirretty esx2-palvelimelle. Kesti noin viisi minuuttia, ennen kuin esx2 hälytti korkeasta prosessorikuormasta (kuva 17) ja DRS alkoi siirtää osaa kuormasta esx1-palvelimen ajettavaksi. (Kuva 18) Siirto tapahtui ensin kahdelle virtuaalikoneelle ja hie- man myöhemmin kolmannelle. Siirron jälkeen molemmat palvelimet ajoivat kuormaa täydellä teholla. Kuvassa 19 näkyy graafisena viivadiagrammina palvelinten prosessorikuorman käyttäytyminen DRS:n siirrettyä osan kuormasta esx2:lta esx1:lle.

```

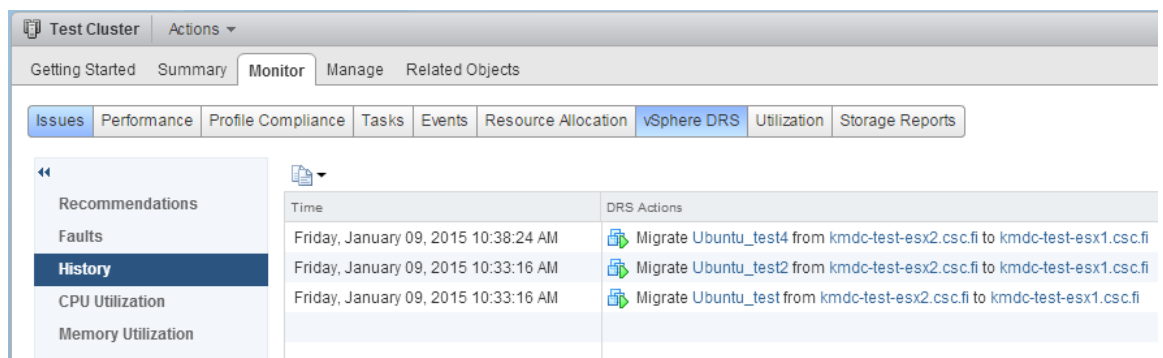
test@kmdc-ubuntu-test1: ~
Ubuntu 14.04 64bit with Linux 3.13.0-32-generic on ktest@kmdc-ubuntu-test1:~$ burnP6
CPU    100.0%   Load  2-core   Mem    27.0%   active:
user:   97.9%   1 min:  1.93   total: 3.86G   inactive:
system: 2.1%    5 min:  1.09   used:  1.04G   buffers:
idle:   0.0%   15 min:  0.70   free:  2.82G   cached:

```

Kuva 16. Kaksi *cpuburn* prosessia ja *glances*-monitori käynnissä virtuaalisessa Ubuntussa

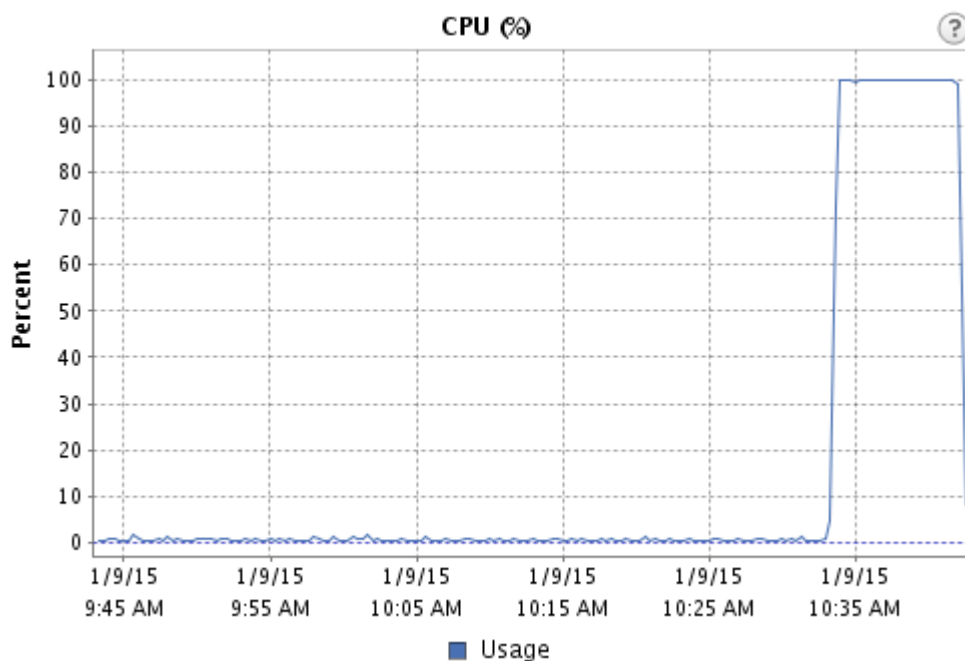


Kuva 17. Noin 5 minuutin kuormituksen jälkeen esx2 hälytti korkeasta CPU kuormasta

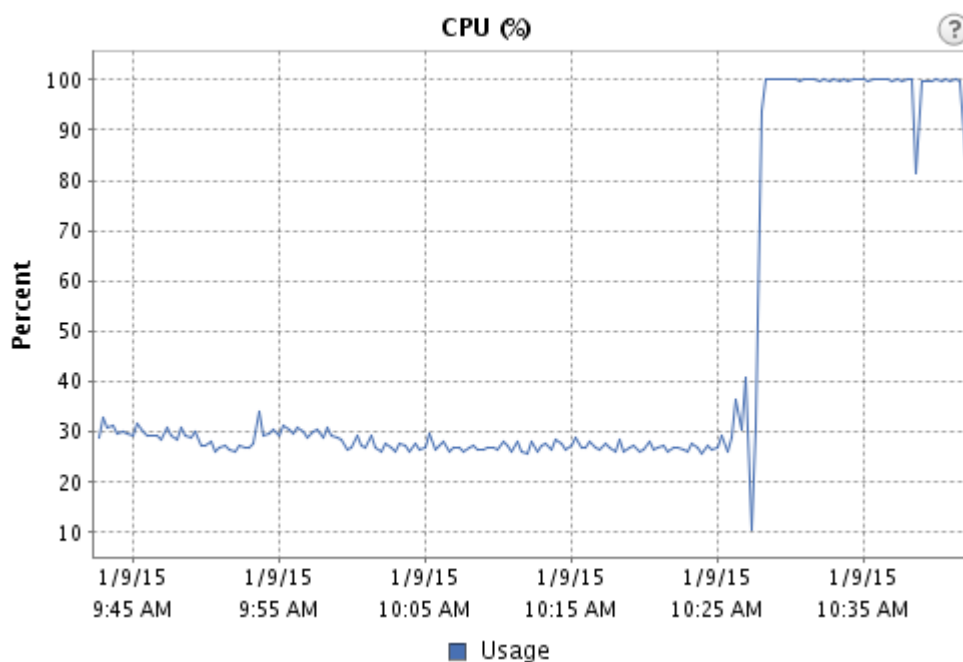


Kuva 18. DRS tasasi kuormaa siirtämällä kolme testikonetta toiselle palvelimelle

Realtime Summary for kmdc-test-esx1.csc.fi



Realtime Summary for kmdc-test-esx2.csc.fi



Kuva 19. Palvelinten prosessorikuormitus DRS:n toimiessa

Testin loppupäätelmänä todettiin dynaamisen resurssien tasauksen toimivan niin kuin pitääkin. Aluksi toinen palvelimista ajoi kaikkia viittä täydellä prosessorikuormalla toimivaa virtuaalikonetta yksin, mutta hetken päästä järjestelmä siirsi ensin kaksi ja lopuksi vielä kolmannen virtuaalikoneen aiemmin tyhjänä olleelle palvelimelle. Siirron jälkeen kumpikin palvelin ajoi

kuormia täydellä teholla, mutta koska kuormat olivat jakautuneet kahdelle palvelimelle, tuli kullekin virtuaalikoneelle enemmän suorituskyyä.

#### Tallennusyhteyden failover-testi

Levyjärjestelmäyhteyden vikasietoisuutta testattiin sulkemalla kytkimen päästä levyliikenteen käyttämä portti ja seuraamalla, kuinka sujuvasti järjestelmä kääntää liikenteen rinnakkaiselle yhteydelle. Testi ajettiin kaksi kertaa, jotta nähtiin kuorman siirtyminen kummaltakin yhteydeltä toiselle.

Testiä varten luodulle virtuaaliselle Ubuntu Linuxille luotiin levylle raskasta levykuormaa *iozone*-ohjelmalla. Ohjelma kirjoittaa testitiedostoon mahdollisimman nopeasti ja vuoroin lukee luotua tiedostoa. Käynnistyskomento testiä varten oli: *iozone -i 0 -i 1 -s 10G -r 1M -f /tmp/testifile*. Samaan aikaan monitoroitiin levyn vasteaikaa *iostat*-nimisellä ohjelmalla sekunnin välein tehtävillä tarkistuksilla. Ohjelma käynnistettiin parametrilla *iostat -x 1*. Ohjelma laitettiin kirjoittamaan monitorointidataa lokitiedostoon. Samalla seurattiin levyjärjestelmän hallinnasta, vaihtuuko I/O-kuormaa tekevä IP-osoite. Testin kanssa yhtä aikaa ajettiin myös *vmkping*-ohjelmaa palvelimen ja levyjärjestelmän välisten katkosten varalta.

Ensimmäinen testitapaus näkyi *iostat*:in lokitiedostossa kolmen sekunnin aikana ensin levyliikenteen pysähtymisenä ja sen jälkeen kasvaneena vasteaikana. Toisen testitapauksen aikana tapahtuma oli lyhempi ja näkyi vain yhden mittauspisteen aikana. Vasteajat mikrosekunteinä näkyvät taulukossa 1. *Vmkping*-ohjelmassa liikenteen vaihtuminen toiselle yhteydelle näkyi vain kasvaneena viiveenä yhden mittauksen aikana (kuva 20). Toisen testitapauksen aikana *vmkping*:issä katosi reilusti paketteja (kuva 21) ja levyjärjestelmän hallinnasta hävisi näkyvistä molemmat kyseisen palvelimen yhteydet. Testikuormaa tekevän virtuaalikoneen lokitiedostossa tämä katkos näkyi kuitenkin pienempänä kuin ensimmäinen testitapaus. *Iostat*:in vasteaika kasvoi vain yhden mittauspisteen aikana. Virtuaaliympäristön puolella yhteyden katkeaminen näkyi kummallakin kerralla levyjärjestelmän iSCSI-adapterin kahdennuksen häviämisenä (kuva 22).

```

64 bytes from 10.220.0.100: icmp_seq=32 ttl=255 time=0.157 ms
64 bytes from 10.220.0.100: icmp_seq=33 ttl=255 time=0.147 ms
64 bytes from 10.220.0.100: icmp_seq=34 ttl=255 time=0.154 ms
64 bytes from 10.220.0.100: icmp_seq=35 ttl=255 time=0.191 ms
64 bytes from 10.220.0.100: icmp_seq=36 ttl=255 time=0.145 ms
64 bytes from 10.220.0.100: icmp_seq=37 ttl=255 time=0.144 ms
64 bytes from 10.220.0.100: icmp_seq=38 ttl=255 time=1.257 ms
64 bytes from 10.220.0.100: icmp_seq=39 ttl=255 time=0.159 ms
64 bytes from 10.220.0.100: icmp_seq=40 ttl=255 time=0.563 ms
64 bytes from 10.220.0.100: icmp_seq=41 ttl=255 time=0.347 ms
64 bytes from 10.220.0.100: icmp_seq=42 ttl=255 time=0.378 ms

```

Kuva 20. Ensimmäisen testitapauksen aikana esiintynyt viiveen kohoaminen näkyy vmkpingin tulosteessa. Ei kadonneita paketteja

```

---          ping statistics ---
101 packets transmitted, 42 packets received, 58% packet loss
round-trip min/avg/max = 0.109/0.145/0.381 ms
~ #

```

Kuva 21. Toinen testitapaus. Paketteja katosi paljon

Adapter Details				
<div> <div>Properties</div> <div>Devices</div> <div>Paths</div> <div>Targets</div> <div>Network Port Binding</div> <div>Advanced Options</div> </div>				
<div> <div>+</div> <div>×</div> <div>i</div> </div>				
Port Group	VMkernel Ad...	Port Group Policy	Path Status	Physical Network Adapter
iSCSI1 pg (Storage...)	vmk1	Compliant	Last active	vmnic4 (10 Gbit/s, Full)
iSCSI2 pg (Storage...)	vmk2	Compliant	Not used	vmnic6

Kuva 22. Yhteyden katkettua iSCSI-adapterin verkkoporteista toinen näkyi Not used -tilassa

Taulukko 1. *iostat*-ohjelman vasteajat mikrosekunteina

Testitapaus 1		Testitapaus 2	
Aika: (s)	await	Aika: (s)	await
1	1393,06	1	2252,61
2	1521,51	2	2163,32
3	2053,92	3	2121,45
4	0	4	2297,89
5	36914,39	5	23922,19
6	37781,94	6	824,48
7	2295,75	7	1618,89
8	1829,31	8	2282,03
9	2178,16	9	2216,67
10	2165,05	10	2223,06

Testin tuloksena todettiin tallennusyhteyden kahdennuksen toimivan niin kuin kuuluukin, eikä toisen yhteyden katkosta tule käyttäjälle kuin korkeintaan pieni hidastuminen luku- tai kirjoitusoperaatioihin.

### 5.3 Käyttöönotto ja ylläpito

Järjestelmä otettiin käyttöön CSC:llä viimeisten testien suorittamisen jälkeen. Testien myötä järjestelmän asetukset todettiin määritykset täyttäviksi ja järjestelmä voitiin ottaa testauskäyttöön. Käyttöön otetussa järjestelmässä oli valmiina viisi asetuksiltaan identtistä Ubuntu Linux 14.04 -virtuaalikoneita, jotka luotiin testien suorittamista varten. Näitä koneita voidaan käyttää yleisesti tuleviin testaustapauksiin. Lisäksi järjestelmään tehtiin muitakin virtuaalikoneita: Open SUSE 13.2 ja Fedora Linux 21.5.

Levyjärjestelmän ”Images datastore” -levyalueelle tehtiin hakemistot niin Linux- kuin Windows-käyttöjärjestelmäversioille. Näihin hakemistoihin on tarkoitus kerätä asennustiedostoja eri käyttöjärjestelmäversioista, jotta testien suorittamiseen olisi aina erilaisia käyttöjärjestelmiä nopeasti saatavilla.

Järjestelmästä on tehty dokumentaatio tilaajan dokumentointijärjestelmään liitettäväksi. Dokumentaatio sisältää järjestelmän tekniset tiedot, verkkoasetukset, käyttäjätunnukset ja ohjeet järjestelmää käyttämiseen. Dokumentaation avulla järjestelmän ylläpito helpottuu. Järjestelmän ja dokumentaation ylläpitoa jatketaan aktiivisesti ja järjestelmää päivitetään käyttötarpeiden muutosten mukaan.

## 6 POHDINTA

Työ lähti käyntiin aika hitaasti, sillä ympäristöä varten tarvittavan laitteiston hankinta vaati aika paljon selvittelyä. Ison yrityksen byrokratia ja hankintakäytännöt veivät oman aikansa ja työn alkuvaiheet menivätkin teoriaan tutustuen. Loppujen lopuksi laitevalintoihin ja asennuksiin ei päässyt vaikuttamaan itse laisinkaan, vaan niistä vastasi CSC:n ryhmä, joka hoitaa IT-hankintoja ja ylläpitää tietojärjestelmiä.

Laitteiden hankinnan jälkeen, kun järjestelmää pääsi käytännössä rakentamaan, kävi työ huomattavasti mielenkiintoisemmaksi. Työn aikana tuli odotetusti ongelmia ja mutkia matkaan. Laitteiden saannin jälkeen joutui niihin tilaamaan uusia tuulettimia, koska komponenttipäivitysten myötä testausympäristöön vapautuneista koneista jouduttiin siirtämään tuulettimia tuotantoympäristön koneisiin. Työn edetessä kävi ilmi, ettei teoriaa opiskellessa ollut osannut huomioida joitakin järjestelmään kuuluvia osia. Tällaisia oli distributed switchin ja iSCSI-verkon kahdennuksen asetusmäärittelyiden suunnittelu etukäteen. Valmiille järjestelmälle suoritettavien käyttöönottestien suunnittelu oli jäänyt kokonaan huomioimatta. Testit onnistuivat kuitenkin hyvin, vaikka niiden suoritustarve huomattiin vasta aivan projektin loppupuolella.

Työn aikana selvisi, että toimeksiantajalta löytyvät jo valmiiksi tämän järjestelmän toteuttamiseen vaadittavat lisenssit VMwaren ohjelmistoihin. Tämä muutti työtä siinä mielessä, että lisenssien hintojen ja vaihtoehtoisten tuotteiden tutkiminen jäi pois. Testausympäristö rakennettiin työn aikana käyttäen testilisenssejä, mutta lisenssit ovat valmiina, kunhan järjestelmä halutaan lisensoida käyttöön.

Järjestelmän palvelimille oli tarkoitus asentaa Dellin OpenManage Server Administrator -ohjelmisto laitteiston monitorointia varten. Vasta ohjelman asennuksen jälkeen kävi ilmi, että kyseinen ohjelma vaatii lisäksi erillisen Windows Serverin internetkäyttöliittymää varten, eikä tietoja saakaan näkyviin vCenteriin. Tässä vaiheessa alettiin selvittämään muuta vaihtoehtoa, jolloin löydettiin Dellin OpenManage Integration -työkalu. Tämä sopi tarkoitukseen huomattavasti paremmin, sillä työkalu lisäsi laitteiston monitoroinnin suoraan vCenteriin ja sen asennus virtual applianceksi oli yksinkertaista.

Mikäli laitteiden hankintapäätökset olisivat olleet opinnäytetyön tekijän päätäntävällässä, olisi järjestelmään hankittu oma levyjärjestelmä. Lisäksi projektin aikana olisi pitänyt perehtyä tar-



kemmin eri teknologioihin ja suunnitella järjestelmän hyväksymistestaus paremmin. Työ onnistui kuitenkin hyvin ja valmis järjestelmä läpäisi sillä suoritettut testit moitteetta. Opinnäytetyöprosessi on lisännyt opinnäytetyön tekijän ammatillisia taitoja järjestelmän kehityksen ja testaamisen osalta. Työn päätavoite oli rakentaa CSC:lle toimiva ja helposti käytettävä testausympäristö.

Järjestelmällä on jo suunniteltu tehtäväksi esimerkiksi palomuurien läpäisytestausta ja failover-testejä erilaisia asetuksia käyttäen. Järjestelmän dokumentaatio julkaistaan tilaajan dokumentointijärjestelmässä ja järjestelmän käyttöä varten pidetään pienimuotoinen koulutustilaisuus Kajaanin toimipisteen henkilökunnalle. Järjestelmää tullaan ylläpitämään ja jatkokehittämään edelleen käyttötarpeiden mukaan.

## LÄHTEET

- Beerens, I. 2013. Allocate a static IP address to the VMware vCenter Server Appliance (VCSA). Viitattu 30.12.2014. <http://www.ivobeerens.nl/2013/01/14/allocate-a-static-ip-address-to-the-vmware-vcenter-server-appliance-vcsa/>
- Crunchbase. Viitattu 11.1.2015. <http://www.crunchbase.com/organization/vmware>
- CSC 2014a. CSC Yhtiönä. Viitattu 9.10.2014. <https://www.csc.fi/csc>
- CSC 2014b. Historia. Viitattu 9.10.2014. <http://datakeskus.csc.fi/historia>
- CSC 2014c. Strategia. Viitattu 9.10.2014. <https://www.csc.fi/strategia>
- CSC 2014d. Datakeskus Kajaani. Viitattu 10.10.2014. <http://datakeskus.csc.fi/datakeskus-kajaani>
- Dell 2014a. OpenManageIntegration for VMware vCenter. Viitattu 12.1.2014. <http://www.dell.com/support/home/us/en/19/Drivers/DriversDetails?driverId=7Y37P>
- Dell 2014b. VMware ESXi 5.5. Image. Viitattu 22.12.2014. <http://www.dell.com/support/home/us/en/04/Drivers/DriversDetails?driverId=5YC4T>
- Dell 2014c. OpenManage Integration for VMware vCenter Quick Installation Guide for vSphere Web Client Version 2.2 Viitattu 12.1.2015. [http://topics-cdn.dell.com/pdf/dell-mgmt-plugin-for-vmware-center-2.2\\_Deployment%20Guide\\_en-us.pdf](http://topics-cdn.dell.com/pdf/dell-mgmt-plugin-for-vmware-center-2.2_Deployment%20Guide_en-us.pdf)
- Doxer 2014. Viitattu 9.1.2015. <http://www.doxer.org/wp-content/uploads/2012/06/what-is-a-cluster.jpg>
- Funet-yhteydet. Viitattu 5.1.2015. [https://www.csc.fi/documents/10180/144815/FUNET-kartta\\_2013.jpg](https://www.csc.fi/documents/10180/144815/FUNET-kartta_2013.jpg)
- Heiskanen. 2014. Operating engineer. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 30.9.2014.
- Hiltunen 2014. Johdatus tietojärjestelmiin. Viitattu 19.10.2014. [http://www.okol.org/verkko-kurssit/datanomi/tietojarjestelmien\\_kaytto\\_ja\\_kehittaminen/johdatus\\_tietojarjestelmiin/kehittamistyön\\_vaiheet\\_ja\\_elikaarimallit/kehittamistyön\\_vaiheet\\_ja\\_elinkaarimallit\\_asia.htm](http://www.okol.org/verkko-kurssit/datanomi/tietojarjestelmien_kaytto_ja_kehittaminen/johdatus_tietojarjestelmiin/kehittamistyön_vaiheet_ja_elikaarimallit/kehittamistyön_vaiheet_ja_elinkaarimallit_asia.htm).
- Karppinen 2014a. Network specialist. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 30.9.2014.

Karppinen 2014b. Network specialist. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 12.12.2014.

Lynn 2014. RAID Levels Explained. Viitattu 13.1.2015. <http://www.pcmag.com/article/0,2817,2370235,00.asp>

Niemi 2014. Development Manager. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 18.12.2014.

Niinenen 2014a. Systems Specialist. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 30.9.2014.

Niinenen 2014b. Systems Specialist. CSC – Tieteen tietotekniikan keskus Oy. Haastattelu 30.12.2014.

Nuotio 2014. Jälleen uutta supertehoa Kajaaniin. Kainuun Sanomat 3.12.2014. Viitattu 5.1.2015. <http://www.kainuunsanomat.fi/kainuun-sanomat/kainuu/jalleen-uutta-supertehoa-kajaaniin>

Pohjonen. 2002. Tietojärjestelmien kehittäminen. Jyväskylä: Docendo.

Rönty 2014. Sisulla pärjää hetken. Kainuun Sanomat 3.12.2014. Viitattu 5.1.2015. [http://yle.fi/uutiset/sisulla\\_parjaa\\_hetken\\_nyt\\_suomalainen\\_on\\_37\\_nopein\\_maailmassa/7667550?origin=rss](http://yle.fi/uutiset/sisulla_parjaa_hetken_nyt_suomalainen_on_37_nopein_maailmassa/7667550?origin=rss)

VMware 2013a. Best Practices for Running VMware vSphere on iSCSI. Viitattu 30.12.2014. [http://www.vmware.com/files/pdf/iSCSI\\_design\\_deploy.pdf](http://www.vmware.com/files/pdf/iSCSI_design_deploy.pdf)

VMware 2014a. Client interfaces for vSphere. Viitattu 5.1.2015. <https://pubs.vmware.com/vsphere-55/index.jsp#com.vmware.vsphere.vcenter-host.doc/GUID-6710A6C4-00AE-4930-BBD6-BDEF78029994.html>

VMware 2014b. Configuring CHAP parameters for iSCSI Adapters. Viitattu 30.12.2014. <http://pubs.vmware.com/vsphere-51/index.jsp#com.vmware.vsphere.storage.doc/GUID-AC65D747-728F-4109-96DD-49B433E2F266.html>

VMware 2014c. ESXi Networking Security Recommendations. Viitattu 2.12.2014. <https://pubs.vmware.com/vsphere-55/index.jsp#com.vmware.vsphere.install.doc/GUID-603AF714-ABD7-429D-8B93-1701F5A80F64.html>

VMware 2014d. Installing vCenter Server 5.5 best practices. Viitattu 27.11.2014. <http://kb.vmware.com/kb/2052334>

- VMware 2014e. Methods of installing ESXi 5.5. Viitattu 2.12.2014.  
[http://kb.vmware.com/selfservice/search.do?cmd=displayKC&docType=kc&docTypeID=DT\\_KB\\_1\\_1&externalId=2052439](http://kb.vmware.com/selfservice/search.do?cmd=displayKC&docType=kc&docTypeID=DT_KB_1_1&externalId=2052439)
- VMware 2014f. vCenter Server Software Requirements. Viitattu 2.12.2014.  
<https://pubs.vmware.com/vsphere-55/index.jsp?topic=%2Fcom.vmware.vsphere.install.doc%2FGUID-F2C3B932-21BB-4A87-B49C-D988B5D16C52.html>
- VMware 2014g. vSphere Distributed Switch Architecture. Viitattu 11.1.2015.  
<https://pubs.vmware.com/vsphere-55/index.jsp#com.vmware.vsphere.networking.doc/GUID-B15C6A13-797E-4BCB-B9D9-5CBC5A60C3A6.html>
- VMware 2014h. vSphere Standard Switches. Viitattu 11.1.2015.  
<https://pubs.vmware.com/vsphere-55/index.jsp#com.vmware.vsphere.networking.doc/GUID-350344DE-483A-42ED-B0E2-C811EE927D59.html>
- Waters, J. 2007. Virtualization Definition and Solutions Viitattu 11.1.2015.  
<http://www.cio.com/article/2439494/virtualization/virtualization-definition-and-solutions.html>
- Watkins, J. 2001. *Testing IT : An off-the-shelf software testing handbook*. Port Chester, NY, USA: Cambridge University Press.